文部省検定済教科書 | 7 実教 | 工業 067

電気技術ⅡC

実教出版

昭和57年3月31日　文部省検定済　高等学校工業科用

電気技術ⅡC

大河内正陽

実教出版株式会社

監　修

元専修大学教授 工学博士	大河内正陽

編　修

東京工業大学教授 工学博士	長谷川健介
株式会社ＣＳＫ 総合研究所	矢田光治
元電子技術 総合研究所	田中　隆
元東京都立荒川 工業高等学校教諭	石井孝司

協　力

東京都立小金井 工業高等学校教諭	赤沼岩男
元東京都立葛西 工業高等学校教諭	市川　武
東京都立蔵前 工業高等学校教諭	伊藤恭史
元東京都立本所 工業高等学校長	緒方興助
東京都立町田 工業高等学校教諭	熊谷文宏
東京都立八王子 工業高等学校教諭	高野雄三郎
元東京都立蔵前 工業高等学校教諭	寺井正一
愛知県立名南 工業高等学校教諭	人見　昭

目 次

第10章 電子計算機

第11章 自 動 制 御

第**10**章

電子計算機

計算機には，そろばんのように数字による表現を用いて計算を行うディジタル（計数形）計算機と，計算尺のように数を長さなど他の物理量に変えて計算を行うアナログ（相似形）計算機とがある。ふつう，電子計算機あるいはコンピュータとよばれるものは，ディジタル自動計算機のことである。この場合，計算ということばは，単なる数値計算だけを意味するのではなく，もっと一般的な情報処理という意味を含んでいる。このため，電子計算機は，科学技術計算や事務計算ばかりでなく，文献などの情報検索，化学プラントや製鉄所などの制御，銀行のオンライン業務，列車の座席予約など，広い範囲にわたって重要な役割を果たしている。

大形電子計算機

電子計算機のあらまし

この節のねらい　電子計算機は，単なる数値計算だけでなく，情報を比較・判断したり，大量の情報を記憶したりすることができ，かつ情報の処理速度が速いなど，電子式卓上計算器のような単なる計算器とは異なる優れた特徴をもっている。

この節では，まず電子計算機のしくみや使い方などについて，そのあらましを学ぶことにする。

1.　電子計算機の構成と種類

（1）　計算のしくみ

電子計算機による計算のしくみを理解するために，図1のように，人間が筆算を行う場合の動作と比較して考えてみよう。

われわれが筆算を行う場合，

① まず，与えられた計算がどんな種類の問題で，それを計算するにはどのような手順で行えばよいかを考えて決める。

② 計算手順が決まると，計算に必要な数値などの情報が紙に書いてあれば，それを読んで脳に記憶し，場合によっては計算用紙に書き込んで計算を行う。

図1　筆算による計算のしくみ

③ 計算結果は，一般に手で紙に書いて解答として形を整え，計算を終わる。

この場合，読む，記憶する，計算する，書くなど一連の動作は，すべて脳における判断・指令のもとに実行されているが，電子計算機では，個々の基本動作に対する**命令**は外部から与えなければならない。電子計算機に指示を与えるこのような命令の集まりを精密に記述したものを**プログラム**（program）という。すなわち，電子計算機では，①の計算手順は，あらかじめプログラムの形で記憶させておく。そこで，プログラムの最初に，計算に必要な数値すなわち情報を読めという命令があると，**入力装置**から情報が読み込まれて**記憶装置**に入れられる。次に，加算・減算などの計算の命令があると，記憶装置に入っている情報を**演算装置**に入れて計算を行う。計算結果は，書けという命令によって**出力装置**から印字されて出される。なお，これらの計算手順のすべての指令は，**制御装置**でプログラムが解読された後，各装置に与えられる。このように，現在，一般に使用されている電子計算機は，内部に主記憶装置をもっていて，その中にプログラムを記憶して，自動的に計算を実行している。これを**プログラム記憶式計算機**という。このプログラム記憶式計算機によって，電子計算機のプログラムの作成・変更が容易になり，反復が自動的に行えるので，融通性のある利用ができるようになった。この点は，電子式卓上計算器などと大きく異なる電子計算機の優れた特徴の一つである。

（2） 電子計算機の構成

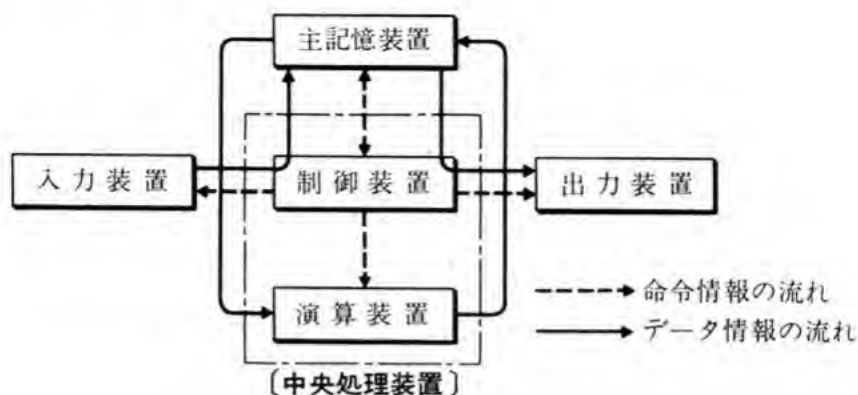
電子計算機は，図2のように，いろいろの装置を組み合わせたものからなっているため，**電子計算組織**とか**計算機システム**とよばれている。なお，命令を解読し，実行する装置を**中央処理装置**（central

小形の電子計算機では，主記憶装置が中央処理装置に含まれているものもある。

図 2　電子計算機の構成

processing unit: **CPU**）という。

中央処理装置が記憶場所を直接指定して情報を書き込んだり，読み出したりできる記憶装置を**主記憶装置**という。また，主記憶装置の記憶容量の不足を補う記憶装置を**補助記憶装置**という。

図 3　電子計算機の構成例

①中央処理装置　②主記憶装置
③カード読取り機　④ラインプリンタ
⑤ディスプレイ装置　} 入出力装置
⑥磁気テープ装置　⑦磁気ディスク記憶装置
⑧フロッピディスク装置　⑨大容量記憶装置　} 補助記憶装置
⑩サブシステム（大形化した電子計算機システム全体を効率よく管理する）

計算機に情報を入力するための入力装置と，計算機から情報を出力するための出力装置をあわせて**入出力装置**という。この入出力装置と補助記憶装置をまとめて中央処理装置に対して**周辺装置**という。

情報には，命令情報とデータ情報がある。計算機をどのように動かすかという動作に関する情報を**命令情報**という。数値情報など，命令情報以外の情報を**データ情報**という。

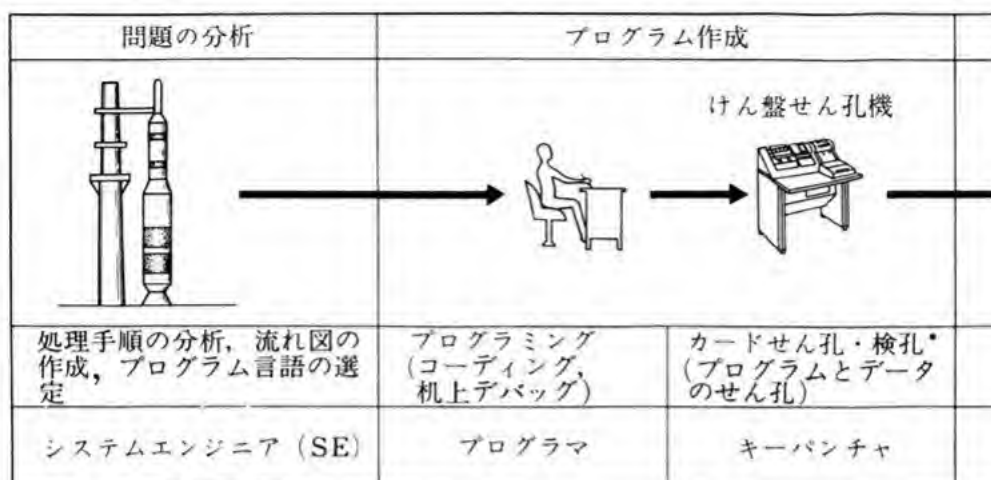
図3は電子計算機の構成例を示したもので，個々の装置については後で学ぶことにする。

電子計算機を構成する装置そのものを**ハードウェア**（hardware）という。また，電子計算機を働かせるための利用技術，すなわち，プログラムを総称して**ソフトウェア**（software）という。

（3） 電子計算機の分類

電子計算機で使用する演算素子は，継電器（リレー）から始まって，真空管・トランジスタ・集積回路（IC）というように開発され，それに伴って小形化され，計算速度は高速化されてきた。電子計算機の実用段階では，真空管による電子計算機を第1世代の電子計算機，トランジスタによる電子計算機を第2世代の電子計算機，集積回路による電子計算機を第3世代の電子計算機という。現在は，**大規模集積回路**（large scale integrated circuit: **LSI**）が大幅に用いられ，第3.5世代とよばれ，さらに超LSIを用いた第4世代へと発展しつつある。

電子計算機は，用途によって事務計算用・科学技術計算用・制御用などに分けられるほか，はん用と専用に大別される。**はん用計算機**は，広い範囲のいろいろな問題を解くことができるように製作されたものである。これに対して，特定業務のために設計され，かつ，完全に単一の作業だけを処理するものを**専用計算機**という。

問題の分析	プログラム作成	
		けん盤せん孔機
処理手順の分析，流れ図の作成，プログラム言語の選定	プログラミング（コーディング，机上デバッグ）	カードせん孔・検孔*（プログラムとデータのせん孔）
システムエンジニア（SE）	プログラマ	キーパンチャ

* せん孔カードの孔が正しくあけられているかどうかを調べること。

また，電子計算機は，価格を尺度にして，超大形・大形・中形・小形・超小形に分けている。* なお，小形で低価格の電子計算機を**ミニコンピュータ（ミニコン）**，1個のLSIで中央処理装置が構成される超小形の電子計算機を**マイクロコンピュータ（マイコン）**とよんでいる。

問 1. 電子計算機はどんな装置から構成されているか。

問 2. 主記憶装置と補助記憶装置の違いを述べよ。

問 3. 電子計算機を用途別に分類せよ。

2. 処 理 手 順

（1） 問 題 の 分 析

電子計算機によって問題を解くには，その問題を解く過程を分析

* JECC コンピューター・ノート（1983年版）によれば，大形2億5000万円以上，中形4000万～2億5000万円未満，小形1000万～4000万円未満，超小形1000万円未満となっている。

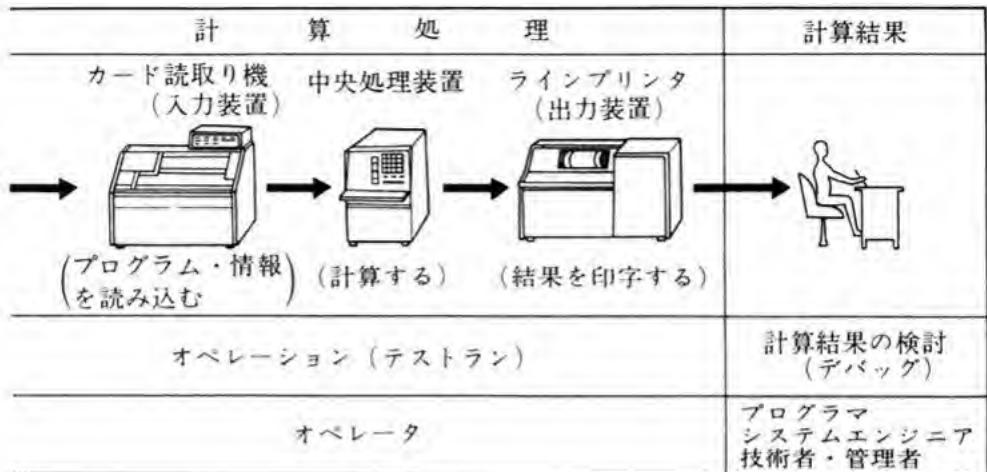



図4 処理手順

し，その個々の過程を電子計算機に対する命令として，一つ一つ実行できる形にする必要がある。そのためには，科学技術計算の場合は，問題の解答に必要な正確に定められた一連の手続き，例えば，ある関数の値を求める数学的な手順の完全な記述が必要になる。このように，問題を解決するための手順を1ステップずつの命令にし，実行できるようにすることを**アルゴリズム**または**算法**という。また，その処理手順を分析し，電子計算機に指令するための手順を図式化して表した**流れ図**(flowchart)とよばれるものを作成したり，プログラムの内容や説明を書いた**仕様書**を作成したりする必要がある。このような作業を行う人を**システムエンジニア**(SE)という。

(2) プログラムの作成

計算処理のための流れ図を作り，それを命令のつながりとして記述するまでを**プログラミング**(programming)といい，プログラムを作成する人を**プログラマ**という。プログラムは，一定の約束に従ったことばで書かれるが，このことばを**プログラム言語**という。また，

流れ図に示された手順をプログラム言語で記述する作業を**コーディング**という。

プログラムは**コーディングシート**とよばれる用紙に記入する。コーディングシートに記入されたプログラムは電子計算機に入力できるように，**けん盤せん孔機**によって，カードにせん孔され，* 同時に，カードの上部に人間が識別できるように文字を印字する。この作業を，**カードせん孔**という。これらの作業を専門に行う人を**キーパンチャ**という。けん盤せん孔機は，ふつうのタイプライタの要領で，キーをたたくことによって，たたいた文字に対応した孔(あな)の組み合わせをカードにあけるものである。

(3)　計　算　処　理

カードにせん孔されたプログラムやデータは，入力装置である**カード読取り機**によって読み取られる。その内容は電気信号に変換され，計算処理の中心となっている中央処理装置に送られて処理される。処理された結果は出力装置，例えば，1行をいちどに印字できる**ラインプリンタ**とよばれる高速の印字装置で印字される。この場合，カードをカード読取り機にセットしたり，電子計算機を始動させたり，計算結果を取り出す作業を行っている人を**オペレータ**という。また，電子計算機が正常に働くように，保守したり，故障を修理したりする人を**カスタマエンジニア**（CE）という。

ラインプリンタで印字された計算結果は，プログラムに誤りはないか，計算結果は正しいかどうかなどが検討され，必要な結果が得られていない場合には，再び繰り返して計算処理が行われる。プロ

---

* 電子計算機への入力には，カードや紙テープ，フロッピディスク，OCRシート，端末機からの直接入力などが用いられるが，ここでは，カードによる入力を考える。

グラムの誤りを見付けて直すことを**デバッグ**という。

このように，プログラムに誤りがないかどうかを確かめるために，プログラムを試験的に実行することを**テストラン**（test run）といい，テストランによって完全なプログラムにしてから，実際のデータ処理が実行される。

問 4. 電子計算機の処理手順に従って装置の名称を挙げてみよ。

問 5. 電子計算機の要員について説明せよ。

問 6. アルゴリズムについて述べよ。

問 7. 次の用語を説明せよ。

(1) プログラミング　(2) コーディング

問　題

1. 次の用語を説明せよ。

(1) 演算装置　(2) 主記憶装置　(3) 制御装置

(4) 入力装置　(5) 出力装置

2. 電子計算機が計算機システムとよばれるのはなぜか。

3. プログラム記憶式計算機にはどんな特徴があるか。

4. ミニコンとは何か。

5. 電子計算機による処理手順を述べよ。

2進数と情報の表現

この節のねらい　電子計算機内部では，数は0と1の組み合わせからなる2進数を用いて表される。これは，計算機内部の機構を考えると，日常使用されている10進数に比べて，電子回路が簡単で，そのため計算の誤りが少ないなどいろいろな利点があるからである。また，データ情報・命令情報なども0と1の組み合わせで表される。

ここでは，2進数と10進数の関係，2進数の計算およびその特徴を調べ，さらに2進数字を使って情報を表す方法などについて調べる。

1．2　進　数

（1）10進数と2進数

電子計算機の内部で用いられている数は**2進法**の数（**2進数**）であるが，われわれが一般に用いている数は**10進法**の数（**10進数**）である。10進数は，10という**基数**を用いるもので，123は，次のように書き表すことができる。

$$123 = 100 + 20 + 3$$
$$\downarrow \quad \downarrow \quad \downarrow$$
$$= 1\times10^2 + 2\times10^1 + 3\times10^0$$

これは，123という数が1個の10^2と2個の10^1と3個の10^0からできていることを意味している。このように，10進数は，0~9までの10個の数字（10進数字）を用いて表し，右の位から10^0, 10^1, 10^2, ……の重みがつけられている。

これに対して2進数は，2を基数として0と1だけで数を表した

ものて，右の位から2^0, 2^1, 2^2, …… の重みがつけられている。したがって，2進数101は次のように表される。

$$101=1\times2^2+0\times2^1+1\times2^0$$

すなわち，2進数の101は10進数の5に相当する。

問 1. 2進数の1011，110010は，10進数で表すといくらか。

（2） 2進-10進変換

2進数を10進数に変換する方法や，逆に，10進数を2進数に変換する方法について調べる。

整数の場合 次の例は，2進数を10進数に変換する例である。

例1 $(11011)_2$を10進数に変換すると，次のようになる。*

$$(11011)_2=(1\times2^4+1\times2^3+0\times2^2+1\times2^1+1\times2^0)_{10}$$
$$=(16+8+2+1)_{10}$$
$$=(27)_{10}$$

表 1 10進数・2進数・16進数の対応関係

10 進 数	2 進 数	16 進 数
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F
16	10000	10
17	10001	11

次の例は，10進数を2進数に変換する例であり，与えられた10進数を2で割っていき，そのつど余りを求め，このように，最後の余りを最上位の数字（most significant digit: **MSD**），最初の余りを最下位の数字（least significant digit: **LSD**）として並べると，変換ができる。

* $(11011)_2$の添字2はかっこ内の数が2進数であることを示し，$(27)_{10}$の添字10は，かっこ内の数が10進数であることを示す。

例2 $(13)_{10}$ を2進数に変換すると，次のようになる。

(商) (余り)

$\frac{13}{2}=6$ 1……… LSD

$\frac{6}{2}=3$ 0

$\frac{3}{2}=1$ 1

$\frac{1}{2}=0$ 1…… MSD

1 1 0 1

$\therefore \quad (13)_{10}=(1101)_2$

小数の場合 2進数の0.1101は，次のように表される。

$$1\times2^{-1}+1\times2^{-2}+0\times2^{-3}+1\times2^{-4}$$

ここで，2^{-1}, 2^{-2}, 2^{-3}, 2^{-4} は $2^{-1}=\frac{1}{2^1}=0.5$, $2^{-2}=\frac{1}{2^2}=0.25$, $2^{-3}=\frac{1}{2^3}=0.125$, $2^{-4}=\frac{1}{2^4}=0.0625$ であるから，次のように変換される。

$$(0.1101)_2=(0.5+0.25+0+0.0625)_{10}=(0.8125)_{10}$$

例3 $(1011.111)_2$ を10進数に変換すると，次のようになる。

$(1011.111)_2$

$=(1\times2^3+0\times2^2+1\times2^1+1\times2^0+1\times2^{-1}+1\times2^{-2}+1\times2^{-3})_{10}$

$=(8+0+2+1+0.5+0.25+0.125)_{10}=(11.875)_{10}$

次に10進数の小数部の変換を例4に示す。これは，例えば10進数の0.625は，次のように表すことができる。

$$(0.625)_{10}=(0.5+0.125)_{10}=(2^{-1}+2^{-3})_{10}=(0.101)_2$$

このことから，10進数，2進数，おのおのを2倍してみると，

$$(0.625)_{10}\times 2=(1.25)_{10}=(0.101)_2\times 2=(1.01)_2$$

$$(0.25)_{10}\times 2=(0.5)_{10}=(0.01)_2\times 2=(0.1)_2$$

$$(0.5)_{10}\times 2=(1.0)_{10}=(0.1)_2\times 2=(1.0)_2$$

となり，10進数，2進数，おのおのを2倍したとき，1の位が等しくなることがわかる。このことから，10進数を2進数に変換するには，例4に示すように行えばよい。

例4 $(0.8)_{10}$ を2進数に変換すると，次のようになる。

$$\begin{array}{cccccc} 0.8 & \to 0.6 & \to 0.2 & \to 0.4 & \to 0.8 & \to 0.6 \; \to \\ \times\;\;2 & \times\;\;2 & \times\;\;2 & \times\;\;2 & \times\;\;2 & \times\;\;2 \\ \hline 1.6 & 1.2 & 0.4 & 0.8 & 1.6 & 1.2 \\ \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ 1 & 1 & 0 & 0 & 1 & 1 \end{array}$$

$$\therefore\quad (0.8)_{10}=(0.110011\cdots)_2$$

上の式で…の部分は，必要な精度まで変換すればよい。したがって，近似的に次のようになる。

$$(0.8)_{10}\fallingdotseq(0.110011)_2$$

例5 $(280.1875)_{10}$ を2進数に変換すると，次のようになる。

まず，整数部分については

（商）（余り）

$$\frac{280}{2}=140 \qquad 0\cdots\cdots \text{LSD}$$

$$\frac{140}{2}=70 \qquad 0$$

$$\frac{70}{2}=35 \qquad 0$$

$$\frac{35}{2}=17 \qquad 1$$

$\frac{17}{2}=$ 8 1

$\frac{8}{2}=$ 4 0

$\frac{4}{2}=$ 2 0

$\frac{2}{2}=$ 1 0

$\frac{1}{2}=$ 0 1…… MSD ……

100011000

$$\therefore \quad (280)_{10}=(100011000)_2$$

また，小数部分については例4と同様にして，

$$(0.1875)_{10}=(0.0011)_2$$

したがって，

$$(280.1875)_{10}=(100011000.0011)_2$$

となる。

問 2. $(110.011)_2$ を10進数に変換せよ。

問 3. $(119.375)_{10}$ を2進数に変換せよ。

(3) 2進数の計算

加 算 $(1011)_2+(1000)_2$ を計算するには，右のわく囲みのようにする。**けた上げ**は，2個の数の，あるけたの加算の結果が，基数2に等しい場合，1けた上のけたに1を加える操作である。

```
  1011
+ 1000
------
 10011
 ↑
(けた上げ)
```

減 算 $(1001)_2-(0101)_2$ を計算するには，右のわく囲みのようにする。**借り**は，2個の数のあるけたの減算の結果が，0より小さい場合，1けた上

```
  1001
- 0101
------
  0100
    ↑
 (借り)
```

から1を引く操作である。

電子計算機では，減算を行うのに減数の補数を作って被減算に加算する方法が使われている。これは，**補数器**を使うと加算回路だけで四則演算ができるので，減算回路はいらなくなること，および補数器の回路は簡単に組むことができるためである。

一般にある数の**補数**は，ある数より1けた上の最小数からある数を引けばよい。例えば，10011の補数は次のようになる。

$$
\begin{array}{lr}
(10011)_2 \text{ の1けた上の最小数}\cdots\cdots & (100000)_2 \\
 & -(10011)_2 \\
\hline
(10011)_2 \text{ の補数}\cdots\cdots & (01101)_2
\end{array}
$$

補数を用いた減算は，例6に示すように，被減数に減数の補数を加え，その答えの最上位のけたを無視すればよい。*

例6　補数による2進数の減算

$(1001)_2-(0101)_2$

① 0101の補数を求める。

② 1001に1011を加える。

③ ②の答えの最上位のけたを無視する。

→0100（答え）

```
①  10000
  − 0101
  ──────
    1011

②   1001
  + 1011
  ──────
   10100
   ↑
③  └無視する

（答え） 0100
```

問 4. 次の2進数を10進数に変換せよ。

(1) 11011　(2) 10010　(3) 1101.1　(4) 1010.1　(5) 1110.1

* このことは，例6に示した減算 1001−0101=0100 を次のように書き換えてみるとわかる。

1001−0101
=1001+(10000−0101)−10000
=1001+1011−10000
=10100−10000
=0100

乗　算　乗算は，加算の繰り返しによって行われる。

除　算　除算は，減算の繰り返しによって行われる。例えば，被除数から除数を引いてみて，もし引けたら商に1をたて，もし引けなかったら商に0をたてる方法を繰り返す。

問 5. 次の2進数の計算を行い，答えを10進数に変換せよ。　　5

(1) 1101＋1111　(2) 101×1110　(3) 1101001÷101

(4) 16 進 数

2進数はけた数が多くなり，読むときや書き表すときに取り扱いが不便なので，16進数に直して扱うことが一般的である。

16進数は，16を基数とした数の表し方で，10進数の10をA，　　10
11をB, …, 15をFで表す。2進数を16進数に変換するには，$2^4=16$であることから，2進数を2^0から4けたずつ区切って，次のように16進数に直す。

$$\begin{array}{ccc} 1010 & 1111 & 0011 \\ \downarrow & \downarrow & \downarrow \\ \mathrm{A} & \mathrm{F} & 3 \end{array}$$

したがって，これは$(\mathrm{AF}3)_{16}$である。　　15

表1 (p. 11) は，2進数，16進数と10進数の関係をまとめて示したものである。

問 6. 10進数の385は，16進数ではいくつか。

問 7. 次の10進数を2進数および16進数に変換せよ。

(1) 387　(2) 781　(3) 111.5　(4) 93.75　(5) 0.625　　20

2. 情報の表し方

(1) 情報の表現

電子計算機内で，文字などの情報は，二つの電気的な符合（コード）の組み合わせで表される。この二つの符号を0と1に対応させると，数字についていえば，10進数を2進数で表すことに相当する。二つの状態を表す方法として，電子計算機では図1のような方法が用いられる。

1）カードや紙テープなどに孔(あな)があいているか，いないか。

2）磁心の磁化の向きが左回りか，右回りか。

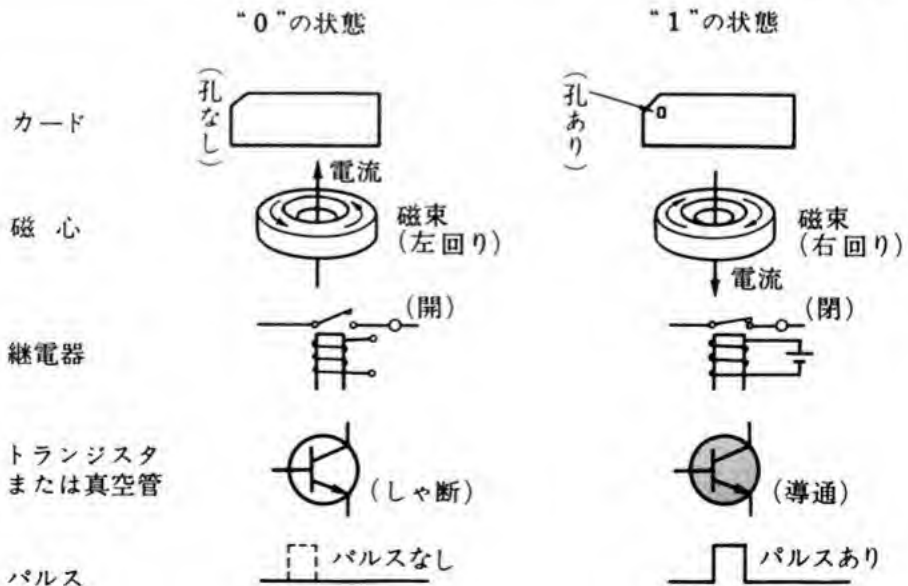
3）継電器やスイッチが開いているか，閉じているか。

4）トランジスタや真空管が導通状態にあるか，ないか。

5）パルスが入ってきたか，こないか。

このように，二つの状態をとることができ，情報を表すための最小の単位となるものを**ビット**（bit）という。すなわち，1ビットは

図1 情報の表現

“0”の状態
“1”の状態
カード
(孔なし)
(孔あり)
磁心
電流
磁束(左回り)
磁束(右回り)
継電器
(開)
(閉)
トランジスタまたは真空管
(しゃ断)
(導通)
パルス
パルスなし
パルスあり

2進数の1けたである。したがって，2進数のけた数はビット数に相当し，3けたの2進数101は3ビット，4けたの2進数1010は4ビットの2進数である。

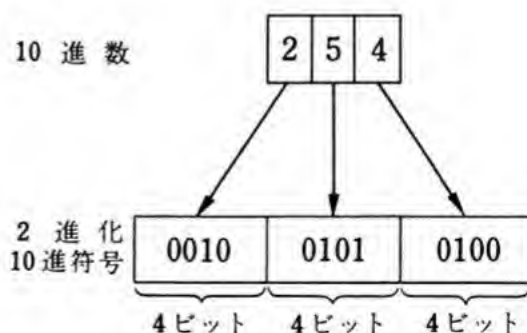
(2) 数字の表し方

電子計算機では，情報を入出力させるとき，10進数を符号で表す方法として，図2のように，各けたの数字をそれぞれ4ビットの2進数で表すことが多い。このようにすると，何けたの数値でも簡単に符号で示すことができる。このように，10進数の1けたを4ビットの2進数によって表した符号を **2進化10進符号** または **BCD** (binarycoded decimal) **コード**という。したがって，図2の例のように，10進数の254は2進化10進符号では001001010100のように表す。

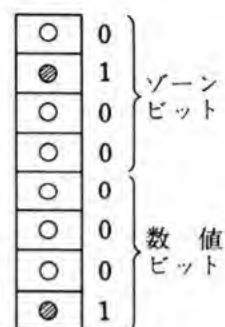
(3) 文字の表し方

電子計算機を使用するときにはプログラムを作るが，プログラムを書くときは，数字のほかに，英字や＋，－などの特殊文字が使われる。そこで，電子計算機内にそのプログラムを入れる場合，電子

図2　2進化10進符号

10進数	2進化10進符号
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001
10	0001 0000
11	0001 0001
12	0001 0010

数　字	符　　号	英　字	符　　号
0	0011　0000	A	0100　0001
1	0011　0001	B	0100　0010
2	0011　0010	:	:　　:
3	0011　0011	:	:　　:
4	0011　0100	O	0100　1111
5	0011　0101	P	0101　0000
6	0011　0110	:	:　　:
7	0011　0111	:	:　　:
8	0011　1000	Y	0101　1001
9	0011　1001	Z	0101　1010

(a)　8単位符号

	ビット	
○	0	ゾーンビット
●	1	ゾーンビット
○	0	ゾーンビット
○	0	ゾーンビット
○	0	数値ビット
○	0	数値ビット
○	0	数値ビット
●	1	数値ビット

(b) 英字Aの符号

数　字	符　　号	英　字	符　　号
0	1111　0000	A	1100　0001
1	1111　0001	B	1100　0010
2	1111　0010	:	
3	1111　0011	:	
4	1111　0100	O	1101　0110
5	1111　0101	P	1101　0111
6	1111　0110	:	
7	1111　0111	:	
8	1111　1000	Y	1110　1000
9	1111　1001	Z	1110　1001

(c)　EBCDIC コード

図3　文字の表し方

計算機内でのこれらの文字の表し方が決められている。

10個の数字だけを表すには，4ビットの長さの符号でじゅうぶんであるが，4ビットでは2^4すなわち16種類の文字しか表すことができない。そこで，数字のほかに英字や+，-などの特殊文字を表すために，4ビットにさらに4ビットを加えた8ビットの長さの符号が用いられる。図3(a)は，JISに定められている**8単位符号**の一部を示したもので，図(b)に示すように，符号の上位4ビットをゾー

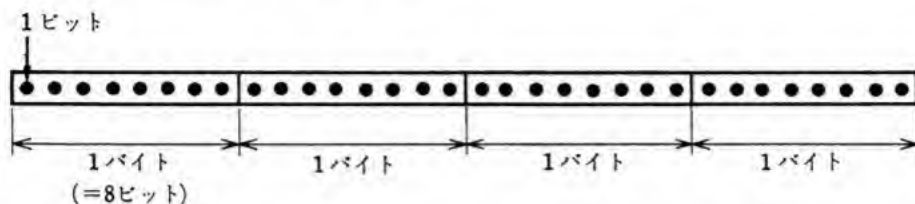



図4 バ イ ト

ンビット，下位4ビットを**数値ビット**という。

8単位符号としては，図(c)に示すような**EBCDIC**(エビシディック)(extended BCD interchange code) とよばれる符号も広く用いられている。

なお，電子計算機内では，情報はこのような8ビットを単位として処理することが多いので，8ビットを1**バイト** (byte) とよんでいる。図4は，バイトとビットの関係を示す。

問 8. ビットとは何か。

問 9. あいうえお……の51文字は何ビットで表現できるか。

問 10. バイトについて説明せよ。

問　題

1. 次の2進数の計算を行え。

(1) 101+111　(2) 1101−1011　(3) 111111×1110

(4) 110111÷101

2. 次の2進数を10進数に変換せよ。

(1) 110110　(2) 101000.11　(3) 1011.01　(4) 1111.101

3. 計算機内では，数字や文字はどのようにして表されているか。

演算回路

この節のねらい　電子計算機内での情報は２進数を用いて表されるが，それらの情報を演算処理するためには，情報を蓄える方法およびその回路や，２進数の演算回路が必要である。

ここでは，その基礎となる論理回路，およびそれを組み合わせた簡単な演算回路について調べる。

1. 論理回路

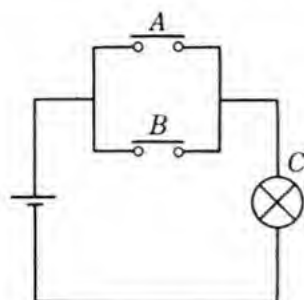
論理代数　電子計算機内では，２進数字の０と１を電気信号の二つの状態に対応させて取り扱う。例えば，０は電流が流れない状態，１は流れている状態というように対応させる。

このような二つの数値を２値といい，２値の範囲内で考えた代数系を**論理代数**という。論理回路は，この論理代数を回路で実現したもので，論理回路を構成する素子を**論理素子**という。論理素子の基本的なものには，次のようなものがある。

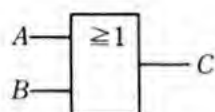
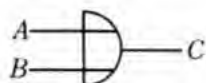
論理和（OR）　論理代数における１けたの数値の和は，次の式で示され，これを論理和という。

$$0+0=0,\ 0+1=1,\ 1+0=1,\ 1+1=1$$

図１(a)で，０をスイッチが開いている状態，ランプが消えている状態に対応させ，１をスイッチが閉じている状態，ランプがついている状態に対応させると，この回路は，論理和 $A+B=C$ を実現していることがわかる。

論理素子の入出力の関係は，図(b)のような真理値表で表すとわ

(a) 接点回路の例

入力		出力
A	B	$A+B=C$
0	0	0
0	1	1
1	0	1
1	1	1

(b) 真理値表

(JISによる表示)　(慣習による表示)　(MILによる表示)

(c) 素子の図記号

図1 論 理 和 (OR)

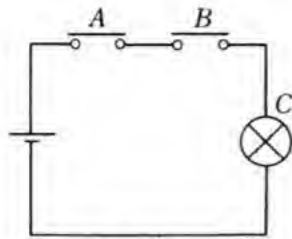
かりやすい。図(b)は論理和素子の真理値表であり，図(c)はその図記号である。

一般に，論理和素子は，二つ以上の入力端子と，一つの出力端子とをもち，入力が一つでも1であれば，出力が1となる性質をもつ。

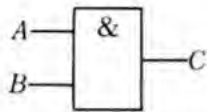
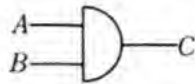
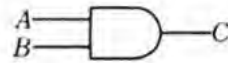
論理積（AND）　論理代数における1けたの数値の積は，次の式で示され，これを論理積という。

$$0\cdot0=0,\ 0\cdot1=0,\ 1\cdot0=0,\ 1\cdot1=1$$

図2(a)で，0，1と，スイッチ，ランプの状態とを，論理和の場合と同様に対応させると，この回路は，論理積 $A\cdot B=C$ を実現していることがわかる。図(b)は論理積素子の真理値表であり，図(c)はその図記号である。

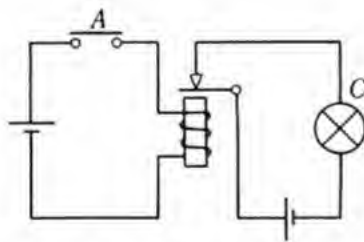
一般に，論理積素子は，二つ以上の入力端子と，一つの出力端子とをもち，入力がすべて1のときだけ，出力が1となる性質をもつ。

(a) 接点回路の例

入力		出力
A	B	$A\cdot B=C$
0	0	0
0	1	0
1	0	0
1	1	1

(b) 真理値表

(JISによる表示)

(慣習による表示)

(MILによる表示)

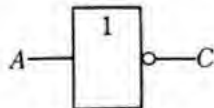
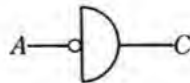
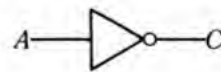
(c) 素子の図記号

図 2 論 理 積 (AND)

(a) 接点回路の例

入力	出力
A	$\bar{A}=C$
0	1
1	0

(b) 真理値表

(JISによる表示)

A C

(慣習による表示)

(MILによる表示)

(c) 素子の図記号

図 3 否 定 (NOT)

否定 (**NOT**)　入力が0のとき出力が1，入力が1のとき出力が0になるような論理素子を否定素子という。

図3(a)は，否定 $\overline{A}=C$ を実現している回路である。図(b)は否定素子の真理値表であり，図(c)はその図記号である。

論理回路に使われる論理素子の基本的なものとしては，以上のほかに，否定と論理積を組み合わせた**否定積** (**NAND**) の素子や，否定

5

図4 否定積 (NAND)

入力		出力
A	B	$\overline{A\cdot B}=C$
0	0	1
0	1	1
1	0	1
1	1	0

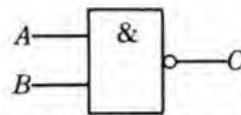
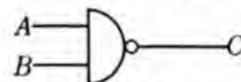
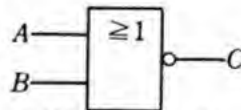
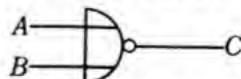
(a) 真理値表

(b) 素子の図記号

図5 否定和 (NOR)

入力		出力
A	B	$\overline{A+B}=C$
0	0	1
0	1	0
1	0	0
1	1	0

(a) 真理値表

(b) 素子の図記号

と論理和を組み合わせた**否定和**（**NOR**）の素子などがある。図4，5はともに，図(a)はそれぞれの論理素子の真理値表であり，図(b)はその図記号である。

問 1. 次の論理式を真理値表を使って表せ。

(1) $A+B+A\cdot B=C$

(2) $(A+B)\cdot A=C$

問 2. 問1の回路を論理素子を使って表せ。

2. 加　算　回　路

2進数の四則演算は，基本的には加算回路を用いて行われる。加算回路は，すでに学んだ論理素子を組み合わせることによって作ることができる。加算回路には，2進数の1けたの加算を行う半加算器と，けた上げも考慮して2けた以上の加算を行う全加算器がある。全加算器は，半加算器を二つ組み合わせることによって作ることができる。

(1) 半　加　算　器

2進数の1けたの加算としては，次の4通りが考えられ，これを真理値表で示すと図6(a)のようになる。

$0+0=0$, $0+1=1$, $1+0=1$, $1+1=0$（けた上げ1）

真理値表において，入力を X, Y, 出力を S, けた上げを C とすると，次のような論理式がなりたつ。

$$S=X\cdot\overline{Y}+\overline{X}\cdot Y, \qquad C=X\cdot Y \qquad (1)$$

上式から出力 S は，X と $\overline{Y}$ の論理積と，$\overline{X}$ と Y の論理積との論理和であり，けた上げ C は X と Y の論理積であることがわかる。

入	力	出	力
X	Y	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

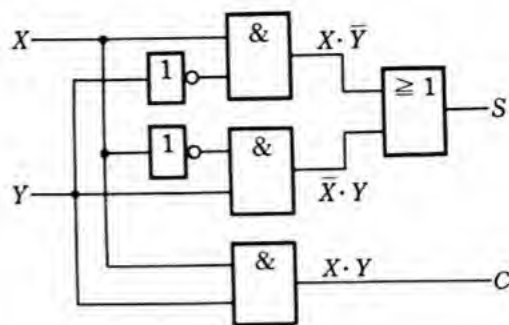
(a) 真理値表

(b) 回　路

図6 半 加 算 器

したがって，これらを論理素子を使って実現すると，図(b)のようになる。これが半加算器の回路である。

（2） 加　算　器

半加算器は，けた上げが入力として考慮されていないので，2けた以上の加算には，けた上げを入力として加える加算器を構成する必要がある。ある1けたの入力を X, Y, 下の位からのけた上げを C_1 とし，出力を S, 上の位へのけた上げを C_2 として，これらの関係を真理値表で表すと，図7(a)のようになり，次のような論理式がなりたつ。

$$\left.\begin{aligned} S&=X\cdot Y\cdot C_1+\overline{X}\cdot Y\cdot\overline{C_1}+\overline{X}\cdot\overline{Y}\cdot C_1+X\cdot\overline{Y}\cdot\overline{C_1} \\ C_2&=X\cdot Y+X\cdot C_1+Y\cdot C_1 \end{aligned}\right\} \quad (2)$$

これらの式を，半加算器の場合と同様に論理素子を使って表すと，図(b)の回路が得られる。これが加算器の回路である。

加算器には，加数を下位のビットから順に計算していく**直列加算器**と，加算器を n 個組み合わせて n けたの2進数の加算が並列にできる**並列加算器**とがある。

問 3. 式(1),(2)の関係がなりたつことを真理値表から確かめてみよ。

入	力		出	力
X	Y	C_1	S	C_2
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

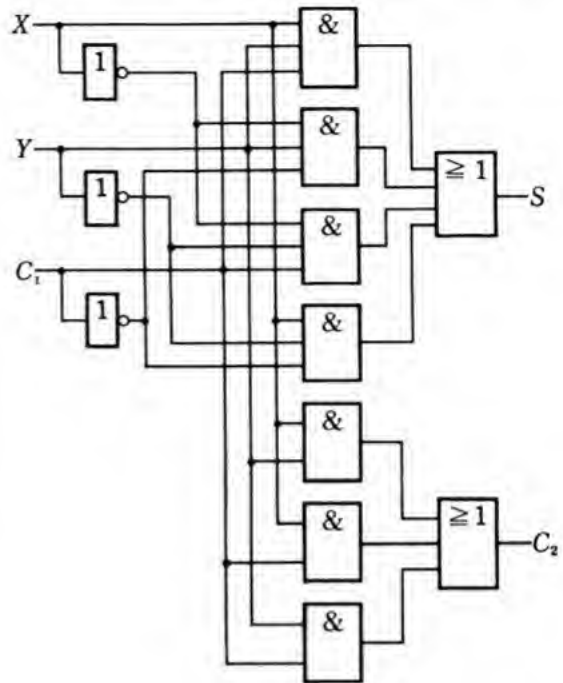
(a) 真理値表

(b) 回 路

図7 加 算 器

3. レ ジ ス タ

(1) 2 値 素 子

何けたかの2進数を電子計算機内で表すには，まずそのうちの1けたについて表すことのできる素子が必要である。このような素子を**2値素子**といい，この素子は0または1の状態を保持することができる。この場合，素子を1の状態にすることを**セット**するといい，0の状態にすることを**リセット**という。

2値素子としては，すでに学んだフリップフロップ回路が用いられる。図8は，フリップフロップ回路を論理回路で示したものである。図において，端子Sが1，端子Rが0になると，Qが1，$\overline{Q}$

(a) 回 路

(b) 図記号

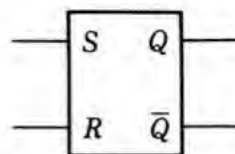
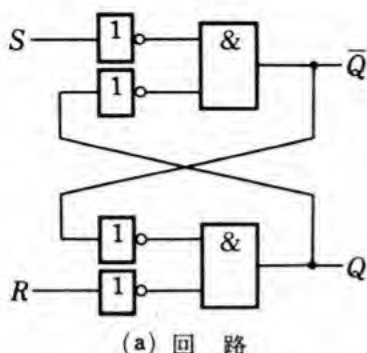



図 8 フリップフロップ回路

は0になり，その後 S が0，R が0になっても $Q=1$，$\overline{Q}=0$ のままその状態を保持する。すなわち，この場合，端子 Q に着目すれば，この素子は2進数の1を表している。逆に，2進数の0を表すには，$R=1$，$S=0$ とすればよい。

このようにして，フリップフロップ回路は1ビットの情報を記憶することができる。したがって，フリップフロップ回路を数個使えば，数ビットの情報を記憶することができる。

問 4. 図8において，$S=1$，$R=0$ のとき $Q=1$，$\overline{Q}=0$ となることを確かめよ。

(2) レ ジ ス タ

レジスタ (register) は，フリップフロップ回路をいくつか接続して，何ビットかのデータを一時記憶し，必要なときには読み出しができるようにした装置である。

図9(a)は，レジスタの働きを示す例として，直列加算器とレジスタの関連を示したものである。すなわち，図における各レジスタは，それぞれの数個のフリップフロップ回路からなり，加数および被加数のレジスタには数けたの2進数が入っている。いま加算が始まると，レジスタ内の数値が1けたずつ加算器に読み出されて加算され，

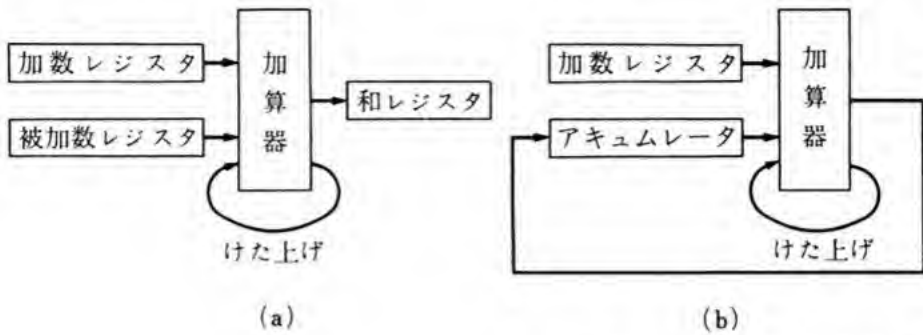



図 9　加算器とレジスタの関係

その結果は順次和レジスタ内に記憶され，加算が終わると最終結果が和レジスタに収められるようになっている。このような加算器を**直列加算器**という。

レジスタ内の数値を1けたずつ読み出すには，図10に示すように，1個のフリップフロップ回路（FF 1）の0か1の状態を他のフリップフロップ回路（FF 2）にけた送り（シフト）することによって行う。

いま，FF 1が1の状態であるとき，送りパルス（シフトパルス）を与えると，図のAND 1はパルスを通すが，AND 2はパルスを通さないので，FF 2のSに1，Rに0が与えられ，FF 2の状態は1になる。このことはFF 1の1の状態がFF 2に移されたことと同じである。

図 10　レジスタのけた送り

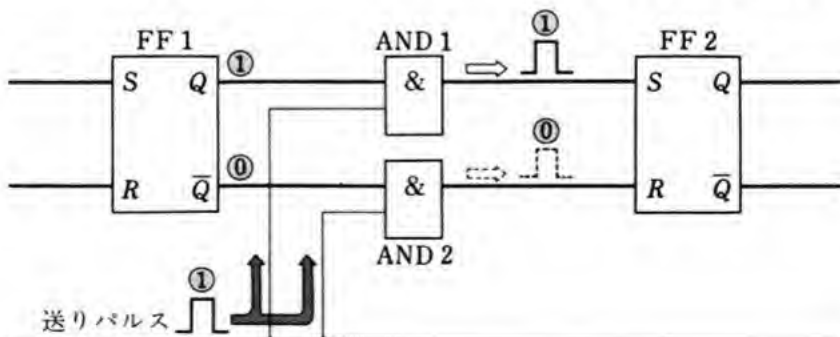
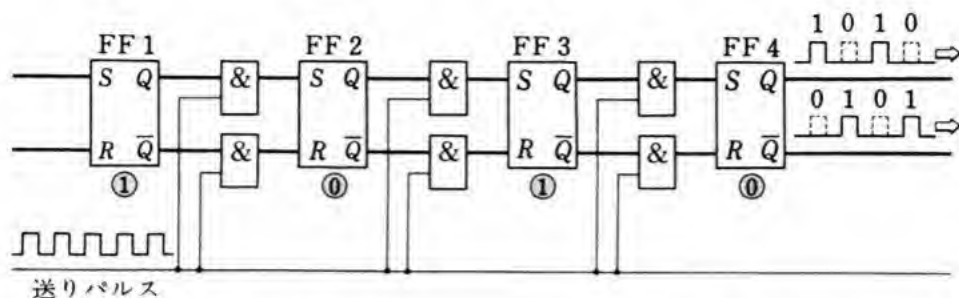



図 11 4ビットシフトレジスタ

ここで使われているANDのように，1個の入力端子，1個の出力端子および制御端子をもち，制御信号によって入力信号がそのまま出力される回路を**ゲート**とよんでいる。

図11は，4ビット **シフトレジスタ** (shift register, **送りレジスタ**)* の例で，送りパルス1個が入ると，1010の状態のレジスタの内容は1010から0101に変わる。したがって，4けた全部を読み出すには，4個の送りパルスを入れれば，図のようなパルス列が，FF 4の Q および $\bar{Q}$ から出ていく。

4. エンコーダとデコーダ

エンコーダ (encoder, **符号器**) は，入力した一つの信号をいくつかの組の信号に直して出力する回路である。例えば，1けたの10進数を入力して4ビットの2進化10進符号を出力するエンコーダを論理回路で示すと，図12(a)のようになる。逆に，複数個の入力信号を解読して，その結果の信号として，ただ一つの出力を作る回路を **デコーダ** (decoder, **復号器**) という。3けたの2進数を入力して

* フリップフロップ回路を何個か継続して，送りパルスを1個与えるごとに，順に次のフリップフロップ回路に情報が転送されるようにしたレジスタをシフトレジスタという。

(a) エンコーダ　(b) デコーダ

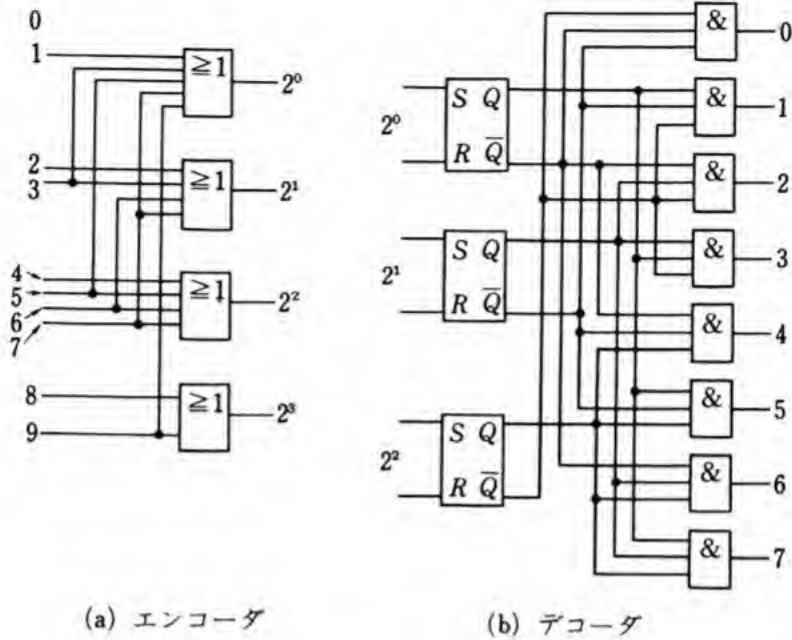



図 12　エンコーダとデコーダ

1けたの10進数を出力するデコーダを論理回路で示すと，図(b)のようになる。

デコーダは，電子計算機内ではいろいろな箇所に用いられている。例えば，命令レジスタ内の命令の種類を解読して，その命令を実行するための制御信号を作るために用いられている。

問 5.　図12(a)で，10進数の入力7が2進数の出力111として得られることを確かめよ。

問 6.　図12(b)で，2進数111が入力として加わったとき，どの出力端子が1となるか。

問　　題

1. 論理回路とは何か。

2. 次の真理値表を完成し，回路名を記入せよ。

(1)

A	B	$C=A+B$
0	0	
0	1	
1	0	
1	1	

(2)

A	B	$C=A\cdot B$
0	0	
0	1	
1	0	
1	1	

(3)

A	B	$C=\overline{A\cdot B}$
0	0	
0	1	
1	0	
1	1	

3. 半加算器と加算器の違いを説明せよ。

4. n ビットの加算を行うには，加算器をどのように組み合わせればよいか考えてみよ。

5. フリップフロップ回路の働きを説明せよ。

6. 加算におけるレジスタと加算器との関係を説明せよ。

7. エンコーダとデコーダを比較し，その働きの違いを説明せよ。

ハードウエア

この節のねらい　第1節で電子計算機のあらましを学んだが，ここでは，さらに演算装置・記憶装置・制御装置のそれぞれのしくみと働きについて調べる。また，現在使われている周辺装置について，その構造や機能を調べる。

1．電子計算機の基本的な概念

（1）命　　令

電子計算機の動作は多種多様である。しかし，これらの動作を分解すると，いくつかの基本動作に分解される。言い換えれば，電子計算機を動かすには，基本動作を次々に指令してやればよい。基本動作を行わせるための情報を**命令**（instruction）という。

中央処理装置と主記憶装置をまとめて，**計算機内部**（machine internal）とよぶことにしよう。計算機内部では，いろいろな情報がやりとりされ，また加工されるが，そのとき，複数個のビットをひとまとめにした**ワード**（word，**語**）という単位で行われる。* ワードに含まれているビット数を**ワードの長さ**（word length）という。多くの計算機では，ワードの長さは一定である。

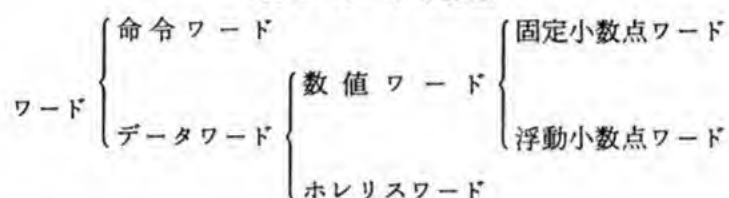
（2）ワ　ー　ド

ワードは，表1に示すように，**命令ワード**（instruction word）と**データワード**（data word）に分けられる。命令ワードは，計算機の命令を表すためのワードである。データワードは，さらに**数値ワード**

* 電子計算機の学習上きわめて大切なことである。

(numeric word) と**ホレリスワード** (Hollerith word) に分けられる。数値ワードは，計算のための数値を表現するためのもので，**固定小数点ワード** (fixed-point word) と**浮動小数点ワード** (floating-point word) に分けられる。固定小数点ワードは，整数を表すワードであり，浮動小数点ワードは，かなり広範囲の有理数を表すワードである。ホレリスワードは，英字・数字・特殊文字を表すためのワードである。

表 1　ワードの分類

- ワード
  - 命令ワード
  - データワード
    - 数値ワード
      - 固定小数点ワード
      - 浮動小数点ワード
    - ホレリスワード

(3)　レコード

計算機内部と，それ以外の部分 (**計算機外部**, machine external), 例えば入出力装置または補助記憶装置との情報のやりとりは，ワード単位ではなく，複数個のワードをひとまとめにした**レコード**(record) という単位で行われる。レコードには2種類ある。一つは**2進レコード**であり，もう一つは**EBCDIC**(エビシディイック)**レコード**とよばれるものである。2進レコードは，主記憶装置と補助記憶装置の情報のやりとりに用いられ，EBCDIC レコードは，一般の入出力に用いられる。

(4)　ブロック

複数個のレコードをひとまとめにしたものを**ブロックレコード** (blocked record) または単に**ブロック** (block) という。大量のデータ処理には，ブロックレコード単位で入出力されることが多い。

2. 主記憶装置

主記憶装置は，ちょうど1ワードの情報を収容できる単位ごとに区切られている。この単位を**ロケーション**（location，**記憶場所**）という。各ロケーションには，逐次的に番号がつけられている。情報の出し入れのときには，この番号でロケーションを識別する。この番号を**アドレス**（address）という。

主記憶装置は，半導体素子によるフリップフロップを使った**半導体記憶装置**，磁気バブルがあるかないかを利用した**磁気バブル記憶装置**，磁心（コア）の選択的な磁性によって情報が記憶される**磁心記憶装置**（magnetic core memory）などがある。

（1） 半導体記憶装置

フリップフロップは，0または1という情報を電気信号として保持する機能をもっている。半導体記憶装置は，このようなフリップフロップの機能をIC化したものである。最近の電子計算機では，ほとんどが半導体記憶装置，つまりIC記憶装置を使っている。

IC記憶装置（IC memory）には，電源が切れたときに，記憶内容が消失する**揮発性記憶装置**と，電源が切れても記憶内容が失われない**不揮発性記憶装置**とがある。

記憶装置の構成　記憶装置は，図1に示すように，記憶部・デコーダ・制御部から構成されている。

記憶部は，図1のように行列に配置された記憶単位から構成されており，アドレス信号の上位アドレス部によって行が，下位アドレス部によって列が選択され，選択された行と列の交点にある1と0の情報を記憶する**記憶セル**が選択されるようになっている。記憶セルの選択は，デコーダによって行われる。

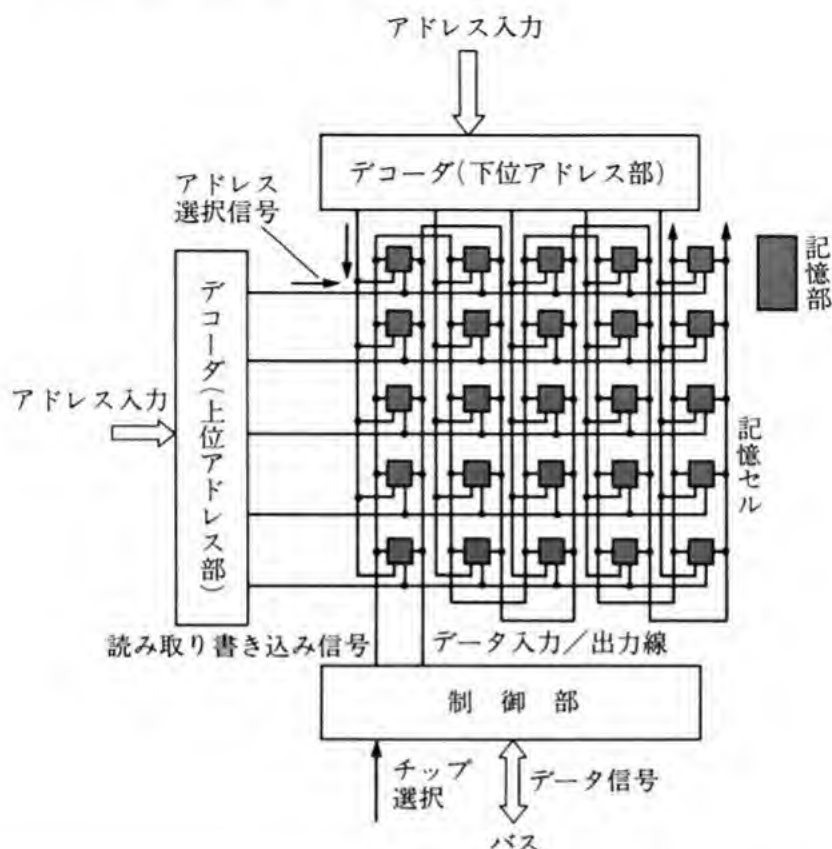



図1 半導体記憶装置の構成

デコーダは，上位アドレス部と下位アドレス部のデコーダからなっており，アドレス信号を解読して，アドレス選択パルスを上位アドレス部と下位アドレス部から記憶部へ送り，任意のアドレスの記憶セルを選択する機能をもっている。

選択された記憶セルは，制御部からの読み取り書き込み信号によって，情報を**読み出す**（記憶セルから情報を取り出す）か**書き込む**（記憶セルに情報を入れる）かが決定される。記憶セルからの情報，または記憶セルへの情報は，データ入力/出力線で，記憶セルから読み出したり，記憶セルへ書き込んだりする。

これらの一連の動作を制御して，記憶セルに情報を記憶させたり，

記憶セルから情報を取り出せるように，CPU（中央処理装置）との情報交換のタイミングを調整するのが**制御部**である。制御部にはこのほかに，**チップ選択**の機能がある。現在，市販されている半導体記憶装置の記憶容量は，64ビットから256Kビットのものがあり，1ワード4ビットから1ワード64ビットまでのものがある。

揮発性記憶装置　このように，任意のロケーションに情報の書き込み，読み取りのできる記憶装置を**RAM**（random access memory，等速呼出記憶装置）という。RAMは記憶セルの構造によって，ダイナミックRAMとスタティックRAMに分けられる。

ダイナミックRAMは，時間とともに記憶内容が失われていくので，記憶装置の内容を保持させておくために，常時再生のための**リフレッシュパルス**（refresh pulse）を供給しておく方式のものである。したがって，周辺回路が複雑になるが，消費電力が少なく，動作速度も速く，集積度も高く，安価なことから，大容量の記憶システムに使われることが多い。

スタティックRAMは，フリップフロップ方式の記憶セルをもったもので，1ビット当たりの消費電力が大きく，動作速度が遅いなどの欠点があるが，わずらわしいリフレッシュやクロックの必要がなく，小容量の記憶装置に使われている。

不揮発性記憶装置　RAMは読み書きのできる揮発性記憶装置であるのに対して，読み出し専用のIC記憶装置として**ROM**（read only memory，固定記憶装置）がある。ROMは，その構造はRAMと同様に，記憶セルの集合マトリックスとアドレスデコーダからなっているが，読み出し専用のため，回路は簡単である。しかし，記憶されている情報は，電源が切れても保持されている不揮発性の記憶装置である。

ROMは，メーカがROMの生産時に用いる**マスク**（記憶情報により異なるパターンをもつ薄い膜）を変えることによって作られる**マスクROM**（mask ROM）と，ユーザが記憶情報を書き込むことができる**PROM**（ピーロム）（programmable ROM，プログラム可能ROM）に大別される。マスクROMは，再び記憶内容が変更できないROMであり，PROMは記憶内容が変更できるものと，できないものとがあるが，とくに変更できるものを**EPROM**（イーピーロム）（electrically programmable ROM，再書き込み可能ROM）という。

（2） 磁気バブル記憶装置

磁気バブルは，非磁性結晶板（オルソフェライト基板）の上に磁性膜を成長させ，その磁性膜の膜面に，垂直な磁化を与えるようにしたものである。これに一定の垂直磁界を外部から加えると，その磁性の膜内に円筒状の磁区を作ることができ，これを**磁気バブル**（magnetic bubble，磁気泡（ほう））とよんでいる。この磁気バブルがあるかないかによって情報の記憶にしている。磁気バブルは，磁性膜内を移動させることができるので，シフトレジスタ形の記憶装置となる。磁気バブルの大きさは2～3〔μm〕くらいであり，チップ上に数十万ビットの情報記憶が可能で，例えば64 K，256 Kビットなどのものがあり，さらに1Mビットのものが開発されている。

磁気バブル記憶装置は，記憶容量の密度が高いこと，また不揮発性記憶装置であるため，動作しないときには電力がいらないこと，機械部分がないことなどの長所がある。

（3） 主記憶装置の性能

電子計算機の主記憶装置の性能には，記憶容量と，記憶の読み出し・書き込みの速度の二つの面がある。

記憶容量は，記憶装置に蓄えることのできる情報の量で，ふつう

バイト数・ビット数で表している。主記憶装置の記憶容量は，小形のもので数百バイト，大形のものでは1億バイト程度に及ぶ。

速度は**呼出し時間**で表示されるが，これは，CPU の制御装置から主記憶装置に対して情報の書き込みや読み取りなどの転送要求があってから，指定されたアドレスをもつロケーションへの転送が開始されるまでの時間である。その後，主記憶装置から情報が読み取られたり，書き込まれたりする。これに要する時間を読出し時間，または書込み時間という。

このように，CPU が主記憶装置を呼び出し，情報を読み出して再書き込みを行い，次の動作が可能になるまでを1**メモリサイクル**という。1メモリサイクルに要する時間を**サイクル時間**という。読出し時間や書込み時間は，呼出し時間より少し余分にかかる程度なので，一般の電子計算機では，サイクル時間は，呼出し時間のほぼ2倍になっている。サイクル時間は1〔μs〕から5.5〔ns〕に及ぶ。

問 1. RAM にはどんな種類があるか。

問 2. サイクル時間とは何か。

3. 周辺装置

(1) カード読取り機・カードせん孔機

カード入出力装置は，カードに孔をあけ，その孔の位置の組み合わせで表された情報を入出力するものである。

カード　ふつう，図2に示したような，横方向に80けた，縦方向に12段のせん孔位置をもつ80けたカードが用いられ，* 各けたに数字・英字または特殊文字の1字分がせん孔される。カードには，上段から12(Y), 11(X), 0, 1, …, 8, 9という位置があって，0以降の位置には番号がつけられている。情報は，図のような簡単な符号で，文字が符号化されている。

カード読取り機　せん孔されたカードの孔の組み合わせを読み取り，カードに表された情報を電子計算機へ送り込む装置で，速度は100～1500〔枚/min〕である。カードの読み取り方式には，機械式と光電式がある。

図2　カードにせん孔された文字

縦 82.6[mm]，横 187.3[mm]

* このほかに，90けたに丸穴をせん孔する90けたカードや，小形電子計算機で使われている96けたカードもある。

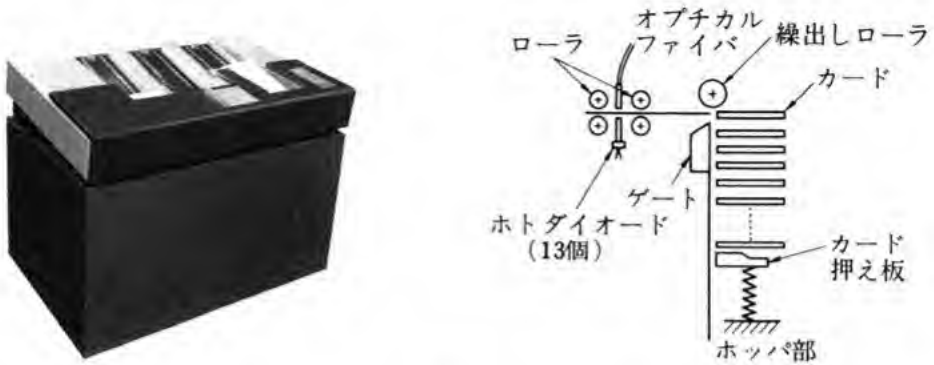



図 3 **カード読取り機**(光電式読取り方式)

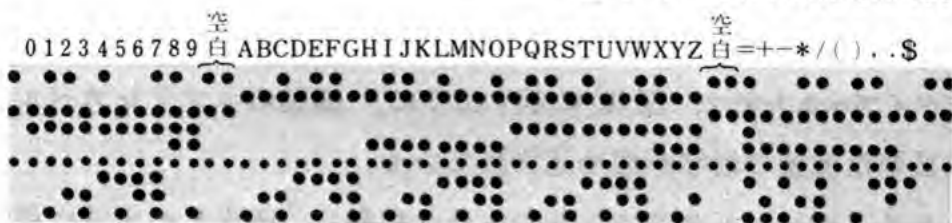
図3の光電式は，カードのせん孔位置に対応させた光電素子と光源の間に，カードを走行させて読み取りを行うもので，高速読み取りができ，信頼性が高いので，一般に多く採用されている。

カードせん孔機　電子計算機で処理された情報を，カードにせん孔するための装置で，速度は100～300〔枚/min〕であり，せん孔されたカードは，後で必要なときに電子計算機の入力媒体として使用できる。

(2) テープ読取り機・テープせん孔機

紙テープ入出力装置は，紙テープに孔をあけ，孔の組み合わせで表された情報を，入出力するものである。

紙テープ　一般に使用されている紙テープは，テープの横方向にせん孔される孔の数が6個の6単位紙テープや，8個の8単位紙

図 4 **テープにせん孔された文字**

テープが多い。図4に示したものは，8単位紙テープの符号である。

テープ読取り機　紙テープ上にせん孔されている情報を読み取って，電子計算機へ送り込む装置である。読み取りの速度は100～1000〔字/s〕である。

テープせん孔機　電子計算機で処理された情報を紙テープにせん孔するもので，せん孔の速度は25～150〔字/s〕である。

(3) タイプライタ

情報が少ない場合や，情報の修正，電子計算機への指令などには，タイプライタが使用され，速度は10～50〔字/s〕前後のものが多い。タイプライタの印字方式には，ふつうのタイプライタと同じ**タイプバー式**，ゴルフボール形に活字を入れて回転しながら選択する**タイプボール式**，ホイールに活字を埋め込んだ**タイプホイール式**や，円筒に活字を入れた**タイプシリンダ式**などがある。

(4) ラインプリンタ

ラインプリンタは，電子計算機の出力結果を印字するもので，1行分の文字をいちどに印字することができ，印字の速度は200～1500〔行/min〕である。

ラインプリンタの活字の組込み機構には，図5のようにタイプドラム式とタイプベルト式があり，1行分の文字はドラムまたはベルトが1回転する時間内に印字されるようになっている。

タイプドラム式は，図5(a)に示すように，円筒上のドラムの外周面に数字・英字・特殊文字・かな文字などの活字が配列されていて，軸方向に1行のけた数に相当する数の活字が並べてある。

タイプベルト式は，図(b)に示すように，鋼製のエンドレスベルトに必要な種類の活字が埋め込まれている。

ドラムやベルトは高速で回転または移動し，印字した文字に相当

(a) タイプドラム式　(b) タイプベルト式

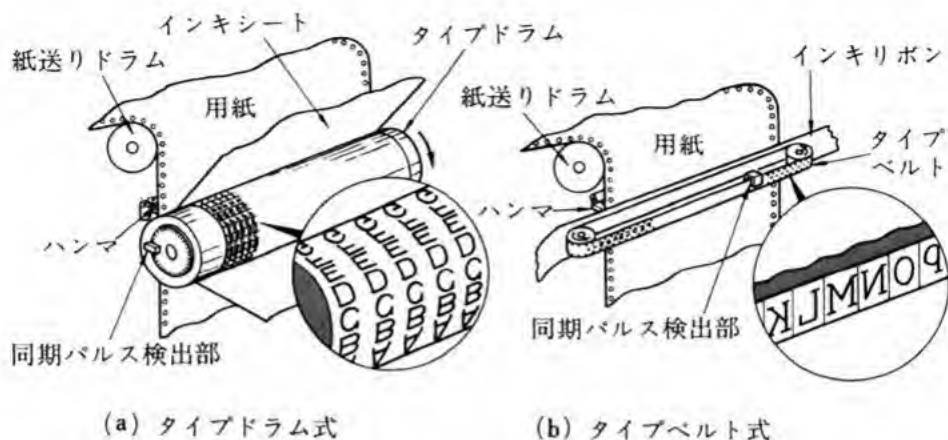



図 5 ラインプリンタの活字組込み機構

する活字がちょうど印字位置にきたとき，印字用紙の裏面からハンマでたたいて印字するようになっている。このため，1行全部が印字される時間はきわめて短い。1行分の印字が終わると，印字用紙は1行分だけ送られ，次の行の印字に移る。

(5) レーザプリンタ

また最近，レーザ光を利用したプリンタも使われている。レーザプリンタは，図6に示すように，プリンタ内の記憶装置に書き込まれた1ページ分の記憶イメージのままのレーザ光を，あらかじめ帯電しておいた感光ドラムに当てると，その部分の静電気だけがなくなる。次に，トナーを感光ドラムと同種の静電気で帯電させ，感光ドラムに付着させると，レーザ光に感光した部分にだけトナーが付着する。そこに連続用紙を当てると，トナーが用紙に付着するので，それを定着させれば，記憶イメージどおりの文字や図形が印刷される。

レーザプリンタは，1000～20000〔行/min〕と高速であるだけでなく，プリンタ内の記憶パターンをそのまま印刷することができるので，図形や日本語のプリンタとして用いられる。

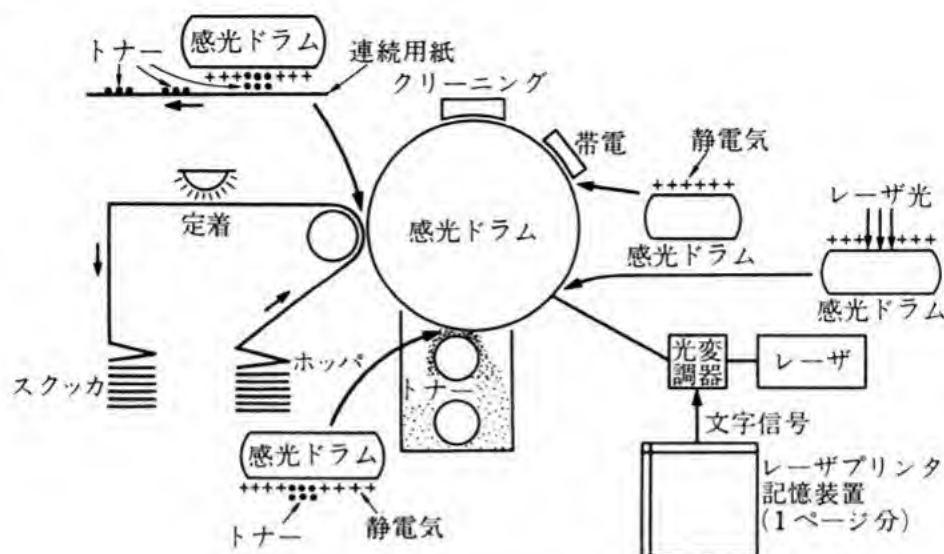
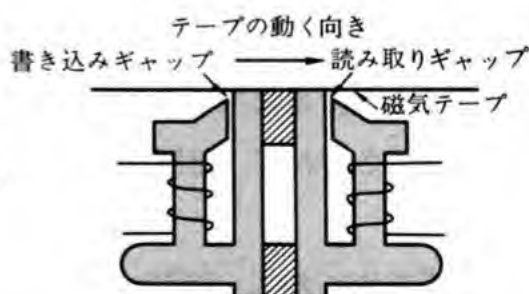



図 6 レーザプリンタ

(6) 磁気テープ装置

磁気テープ装置は，磁気テープ上に符号化された情報の記録を行うもので，テープ1巻(長さ2400フィート(732〔m〕)または1200フィート(366〔m〕)，幅$\frac{1}{2}$インチ(12.7〔mm〕))* には，1500～2500万バイトの大容量の情報を入れることができ，1ビット当たりの価格が安い。

図 7 磁気テープと磁気ヘッド

(a) 磁気テープ　(b) 磁気ヘッド

* 1フィート(feet,〔ft〕)=0.3048〔m〕，1インチ(inch,〔in〕)=25.4〔mm〕

磁気テープ装置の**磁気ヘッド**は，図7(b)のように，パーマロイやフェライトで作られた磁心に巻線が施され，この巻線に流れる電流の向きによって，右回りか左回りの磁界が生ずるようになっている。磁心の一部にせまい磁気ギャップを設けてあり，ここを通る磁束によって，磁性体を磁化するようになっている。

記録は，磁性体の表面に磁気ヘッドを置き，巻線に電流を流して磁性体を飽和するまで磁化することによって行われる。その磁界の向きに磁化された状態が残り，この磁化された部分の磁界の向きによって，"0" と "1" を定義しておけば，一つの部分が1ビットを表すことになる。

再生は次のようにして行われる。磁気ギャップの下を磁化されたテープが通過すると，テープ上の各部分の磁束の向きによって，磁気ヘッドの巻線に誘導電圧が発生する。この誘導電圧によって，記録された情報を読み取ることができる。

なお，テープの幅方向に配列され，記録または再生が同時に行われるビットの1組を**列**という。記録密度はBPI*（または列/mm）で表し，800〔BPI〕(32〔列/mm〕)，1600〔BPI〕(63〔列/mm〕)，6250〔BPI〕(246〔列/mm〕) などがある。

また，磁気テープの長手方向に，1個の磁気ヘッドで読み出し・書き込みができる線状の部分を**トラック**といい，9トラックのものがある。

磁気テープ装置は，比較的安価で，かなり大量の情報を記録することができ，情報をテープ上に順に書き込んだり，読み出したりするのには適している。しかし，いろいろな記憶場所にある情報を，

* BPIはbit per inchの略で，1インチ当たりのビット数を表している。

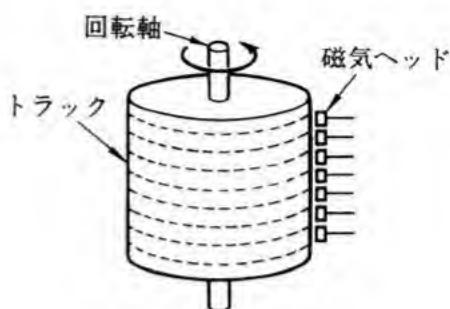
(a) 外 観

(b) 動作原理

図8 磁気ドラム

適宜取り出して処理するには，きわめて不便である。

問 3. 1600〔BPI〕の磁気テープは，1〔cm〕に何バイト記録することができるか。

(7) 磁気ドラム記憶装置

磁気ドラム記憶装置は，図8に示すように，ガンマ酸化鉄やニッケルクロムなどの磁性材料を塗布またはめっきした円筒(ドラム)を，電動機の駆動によって回転させ，微小間隔を保って保持された磁気ヘッドによって，情報の書き込み・読み出しを行う装置である。磁気ヘッドは，トラックごとにふつう1個ずつあり，トラックの選択はヘッドを電気的に切り換えて行う。トラックのうちの何本かは，情報の記録位置を決めるために用いる。

磁気ドラム記憶装置は，いろいろな記憶場所にある情報を，順序によらず処理するのに適し，平均呼出し時間も短い。しかし，同じ記憶容量に対する価格は，磁気ディスク記憶装置よりも一般に高い。

(8) 磁気ディスク記憶装置

磁気ディスク記憶装置は，ガンマ酸化鉄やニッケルコバルトを塗布またはめっきした金属円板（ディスク）の面上に，図9(b)に示す

(a) ディスクパックの外観　(b) 動作原理

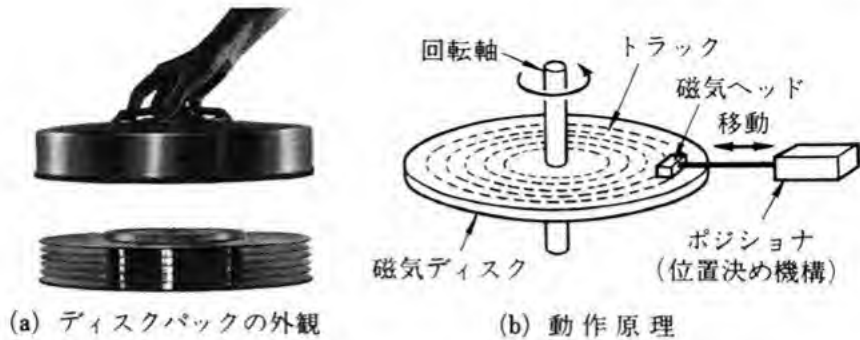



図9 磁気ディスク

ように，磁気ヘッドを置き，同心円状の多数のトラック*に情報を記憶する装置である。ふつう，何枚かのディスクを上下に積み重ねる構造になっている。なお，ディスクを1個ずつ容器に収め，装置に着脱してディスクの交換を可能にしたものを**磁気ディスクパック装置**という。図9(a)は，ディスクパックの外観を示したものである。

ディスクの直径は356～430〔mm〕，1面当たりのトラック数は100～200本，1面当たりの記憶容量は数十万バイト程度であるが，大形のものでは6億バイトに及ぶものもある。

磁気ディスク記憶装置は，いろいろな記憶装置に入っている情報を順序によらず処理するのに適している。

(9) 文字読取り装置

光学式文字読取り装置 (optical character reader: **OCR**) は，文字を光学的方法によって直接読み取る入力装置である。そのための字体として JIS には図10(a)の OCR-A と図(b)の OCR-B の2種類があり，OCR-A は，機械が読みやすいようにした字体である。OCR-

* 図9(b)で，磁気トラックは，わかりやすいように数本しかかいてないが，実際には数十本から数百本ある。トラックとトラックの間隔は，数ミリメートル程度である。

ABCDEFGHIJKLM
NOPQRSTUVWXYZ
0123456789
·⑁:⑂=+/$*"&|
'-{}%?⑀⑁⑂
£¥

(a) OCR-A

ABCDEFGH abcdefgh
IJKLMNOP ijklmnop
QRSTUVWX qrstuvwx
YZ*+,-./ yz m
01234567 £$:;<%>?
89 [@!#&,]¥
(=) "´`^
↑≤≥×÷°¤

(b) OCR-B

図10 光学文字

0123456789⑆⑇⑈⑉

図11 磁気インキ文字

Bは，将来の読み取り技術の発展を見越して，原理的な読みやすさを確保するとともに，美的にも満足すべき字体で，広く一般事務用にも使えるようになっている。

磁気インキ文字読取り装置（magnetic ink character reader: **MICR**）は，磁気インキで印字されたものを読み取る装置である。なお，字形としては，図11のような文字がJISで定められている。

(10) マーク読取り装置

マーク読取り装置には，光学式（optical mark reader: OMR）と磁気式（ULP－ML）がある。光学式はふつうの鉛筆で，磁気式は磁性のしんの鉛筆で，棒状のマークをつけたものを読み取る装置である。

(11) ディスプレイ装置

ディスプレイ装置は，情報を文字・グラフまたは図形で視覚的に表示する装置で，CRTによるディスプレイ装置が，一般に多く用

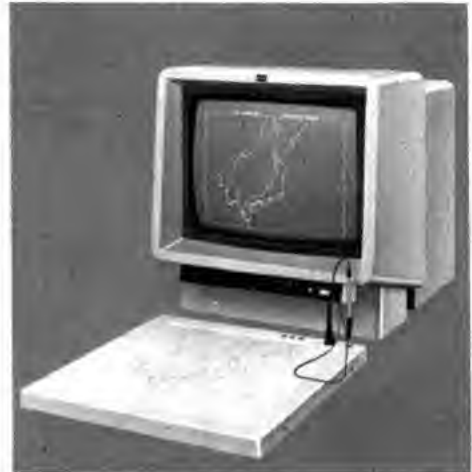
(a) キャラクタディスプレイ装置

(b) グラフィックディスプレイ装置

図 12 ディスプレイ装置

いられている。

CRT ディスプレイ装置は，ブラウン管（CRT）を電子計算機に接続し，ブラウン管上に情報を表示したり，入力したりするもので，図 12 のように，文字を表示する**キャラクタディスプレイ装置**と，図形を表示する**グラフィックディスプレイ装置**とがある。

光電素子を利用して，CRT ディスプレイ装置の画面上に表示された文字または図形に光を当て，文字や図形を修正する制御装置に知らせるものを**ライトペン**という。グラフィックディスプレイ装置では，これを用いて，文字または図形の変更や削除などができる。

(12) X - Y プロッタ

X - Y プロッタは，電子計算機のディジタル量を入力して，その増加量に応じてプロットする機械式の作図機で，ペンが X 軸の向き，Y 軸の向きに動いて平面図形をえがく装置である。

(13) フロッピディスク

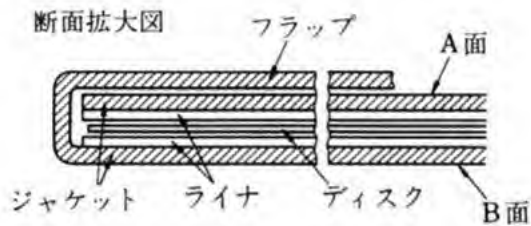
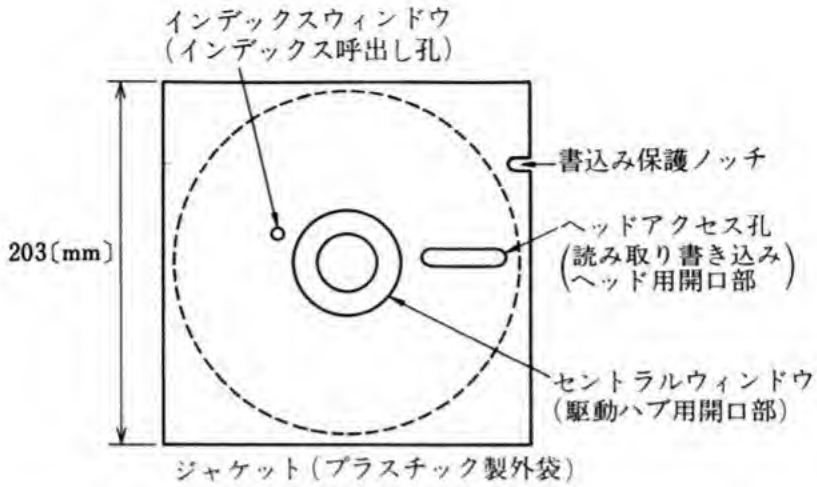
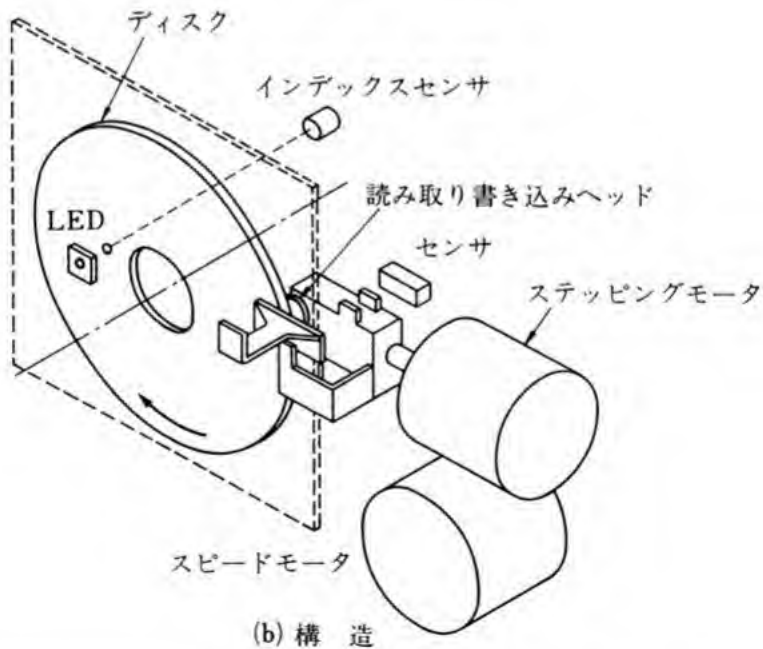
フロッピディスクは，柔軟な円盤とプラスチックの保護ジャケットから構成されており，円盤はジャケットの中で回転するようにな

(a) 外　形

(b) 構　造

図 13 フロッピディスク

っている。円盤の表面には，磁性材料が塗布されていて，円盤上に**トラック**とよばれる同心円状の帯があり，トラックに沿って磁気的に情報を書き込んだり，読み取ったりすることのできる**磁気ヘッド**がついている。フロッピディスクは，曲げることができる磁気ディスクであり，カセットテープやディスクパックと同じように，駆動装置に取り付けたり，取り外して保管や移動をすることができる交換可能形の情報記憶媒体である。

また，通常のフロッピディスクより小形で，直径が13〔cm〕程度のフロッピを**ミニフロッピ**といい，マイクロコンピュータなどで，多く使われている。

問 4. 入力装置・出力装置・補助記憶装置には，どんなものが用いられるか。

問 5. カード読取り機について説明せよ。

問 6. タイプライタの印字方式について説明せよ。

問 7. ラインプリンタについて説明せよ。

問 8. 光学文字について説明せよ。

問 9. グラフィックディスプレイ装置とは何か。

4. 制御装置と演算装置

(1) 命令ワードの構成

われわれがこれから考えていく計算機の命令ワードの構造は，図14のようになっていると想定しよう。

ビットポジション (bit position, **ビット位置**) Sから3までの4ビットは，計算機の基本動作の種類の情報，すなわち**演算コード** (operation code) を表している。ビットポジション4にある情報は，あとで学ぶインデックスレジスタに関係する。これが0であれば，インデックスレジスタを無視することを意味し，1であれば，演算のときにインデックスレジスタの内容を参照することを表している。ビットポジション5から15までの11ビットの情報は，この命令の演算に関連するロケーションのアドレスを意味している。

(2) 演算コード

われわれの計算機の演算コードは，表2のようになっているものとする。

また，これとともに，大切な基本概念であるロードする，ストアするということについて説明しよう。

ロードするとは，一般に，CPUのレジスタに記憶装置，またはCPUの他のレジスタから1ワードの情報を転送することである。

図14 命令ワードの構造

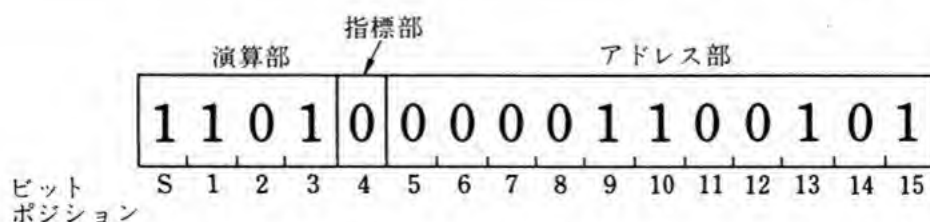



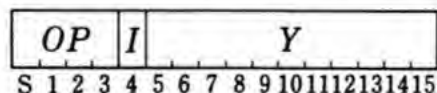
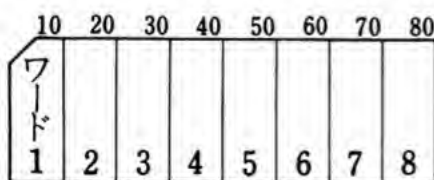
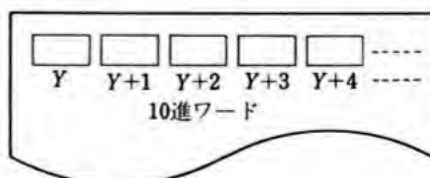
表2 演算コード

コード	演算	略記号	エクセキューションサイクルにおける操作の内容
0000	停止	HLT (Halt)	電算機が停止する。アドレス Y は無視する。
0001	AC クリヤ	ACC (Accumulator clear)	アキュムレータをリセットする。アドレス Y は無視する。
0010	ロード	LOD (Load)	アドレスYをもつロケーションの内容をアキュムレータにロードする。
0011	ストア	STO (Store)	アキュムレータの内容を，アドレスYをもつロケーションにストアする。
0100	加算	ADD (Add)	アドレス Y をもつロケーションの内容をアキュムレータに加算する。その結果の和は，アキュムレータに立つ。
0101	減算	SUB (Substract)	アキュムレーターの内容から，アドレス Y をもつロケーションの内容が引かれる。その結果はアキュムレータに立つ。
0110	乗算	MPY (Multiply)	アキュムレータがクリヤされた状態で，MQ レジスタの内容と，アドレス Y をもつロケーションの内容とが乗ぜられて，結果が拡張された 32 ビットレジスタに立つ。
0111	除算	DIV (Divid)	32 ビットの拡張されたレジスタが，アドレス Y をもつロケーションの内容で除され，結果の商は MQ レジスタに立つ。また，余りがアキュムレータに残る。ただし，アドレスYをもつロケーションの内容の大きさが，アキュムレータの内容の大きさより大きくなければならない。
1000	符号変換	SCH (Sign change)	アキュムレータのSビットは，0が＋で，1が－である。これを逆転させる。アドレス Y は無視する。

コード	演算	略記号	エクセキューションサイクルにおける操作の内容
1001	シフト	SHT (Shift)	アキュムレータの内容を Y 個だけ左へけた送りする。Sから左へ送られたビットは消失する。
1010	ラウンドシフト	RST (Round shift)	アキュムレータの内容を Y 個だけ左へけた送りする。Sから左へ送られたビットはポジション15に移る。
1011	ジャンプ	JMP (Jump)	Y を命令カウンタにロードする。
1100	分岐(ブランチ)	BCH (Branch)	アキュムレータが0であれば，Y を命令カウンタにロードする。0でなければ何も起こらない。
1101	入力	IPT (Input)	図(b)のようなカードを読み込み，10進5けたのワードを2進ワードに変換し，そのワード1をアドレス Y をもつロケーションに，ワード2をアドレス $(Y+1)$ をもつロケーションに，……ワード8をアドレス $(Y+7)$ をもつロケーションに，順次ストアする。
1110	出力	OPT (Output)	アドレス Y をもつロケーションから，連続して10個のロケーションの内容を，10進ワードに変換し，それを1行にプリントする。
1111	無操作	NOP (No operation)	何も起こらない。

OP				*I*	*Y*										
S	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

(a) 命令ワード

10	20	30	40	50	60	70	80
ワード1	2	3	4	5	6	7	8

10進ワード

(b) 10進ワードを読み取る(入力)　(c) 10進ワードで1行に印字する(出力)

ストアするとは，一般に，CPU のレジスタから主記憶装置に 1 ワードの情報を転送することである。

（3） 中央処理装置のレジスタ

われわれの計算機の中央処理装置には，次のようなレジスタがあると考えよう。そのレジスタの名称や働きは，一般的に書かれているが，ビット数は，図 14 に示した命令ワードが用いられるという前提で記述してある。

命令レジスタ (instruction register) は，16 ビットの記憶容量をもつレジスタで，命令ワードを収容するために用いられる。最初の 4 ビットを**演算部** (operation part)，次の 1 ビットを**指標部** (index part)，最後の 11 ビットを**アドレス部** (address part) といい，それぞれ演算コード，インデックス情報，関連アドレスが収容される。

アドレスレジスタ (address register) は，11 ビットの記憶容量をもつレジスタで，命令の実行に必要な情報が記憶されているロケーションを主記憶装置に知らせるために用いられる。

命令カウンタ (instruction counter) は，11 ビットの記憶容量をもつレジスタで，次に実行する命令が収容されているロケーションのアドレスを示している。このレジスタの内容は，通常，命令の実行が終わるたびに，自動的に 1 増加する。

ストレージバッファレジスタ (storage buffer register) は，**メモリバッファレジスタ** (memory buffer register) ともいい，16 ビットの記憶容量をもつレジスタで，中央処理装置と主記憶装置との間の情報のやりとりにさいして用いられる緩衝レジスタである。

以上で，制御装置のレジスタの説明を終える。次に，演算装置のレジスタの説明に移る。

アキュムレータ (accumulator，**累算器**) は，16 ビットの記憶容量を

もつレジスタで，算術演算とか論理演算などのほとんどの演算にかかわりをもっている。いままでのレジスタと違うところは，累算が可能だということである。

MQ レジスタ (Multiplier-Quotient register) は，16 ビットの記憶容量をもつレジスタで，乗除算において特別な機能を発揮する。乗算が行われるとき，アキュムレータと合わせて32ビットの記憶容量のレジスタを形成する。このとき，アキュムレータの最下位のビットが，MQ レジスタの最上位のビットに接続される。

インデックスレジスタ (index register, **指標レジスタ**) は，11 ビットの記憶容量をもつレジスタで，命令のアドレスを変更したり，修正したりするために用いられる。具体的には，行列の演算などを行うときによく用いられる。

(4) マシンサイクル

制御装置は，主記憶装置に記憶されたプログラムの命令を解読して，各装置に必要な情報を送り，命令を実行させる。そのさい各装置の動作の歩調を合わせるのに用いる基本的なパルスを**クロックパルス**といい，必要な装置を動作させるために，制御装置から送るパルスを**指令パルス**という。

マシンサイクル (machine cycle) は，**インストラクションフェッチサイクル** (instruction fetch cycle, **命令取出し段階**) と**エクセキューションサイクル** (excution cycle, **命令実行段階**) とよばれる二つのサイクル (段階) から構成されている。それぞれのサイクルは，複数個のクロックパルスの間隔を基調とする期間からなっている。

制御装置は，命令カウンタが示している 11 ビット情報をアドレスとするロケーションを探す。見付けたらそのロケーションに入っている内容を命令レジスタにロードする。ここまでの段階をインス

トラクションフェッチサイクルという。

次に，制御装置は，命令レジスタの内容の解読を始める。命令レジスタの演算部の4ビット情報を解読して，次に行うべき基本動作を判断して認識する。さらに指標部の1ビットが1であれば，インデックスレジスタを参照しなければならないことを知る。さらにアドレス部の11ビットをアドレスレジスタにロードする。次に，必要な装置に指令を出すと，初めて動作が開始される。実行が終わったら，命令カウンタの内容が自動的に1増加する。これでエクセキューションサイクルが終わる。

一つのインストラクションフェッチサイクルと，一つのエクセキューションサイクルとを合わせて**1マシンサイクル**という。

制御装置では，一つのマシンサイクルが終わると，自動的に次のマシンサイクルが始まるようになっている。このように，電子計算機では，次々にマシンサイクルが繰り返され，いろいろな動作が継続的に起きるのである。

（5） 電子計算機の動作

制御装置は，記憶装置に入っているプログラムを，定められた順序に従って取り出し，その命令を解読して，演算装置や入出力装置に指令を出し，それらの装置を制御するものである。制御装置は，例えば，命令レジスタ・命令カウンタ・デコーダなどで構成されている。

ここで，$c=a+b$ という式の計算を例にとって，図15を参照しながら，電子計算機の動作を説明しよう。まず，入出力，情報の流れ，プログラムの設定は，次のようになっているものとする。

すなわち，a と b に対する数値35，18がそれぞれ図16のようにカードにせん孔されている。この数値をカードから読み取り，それ

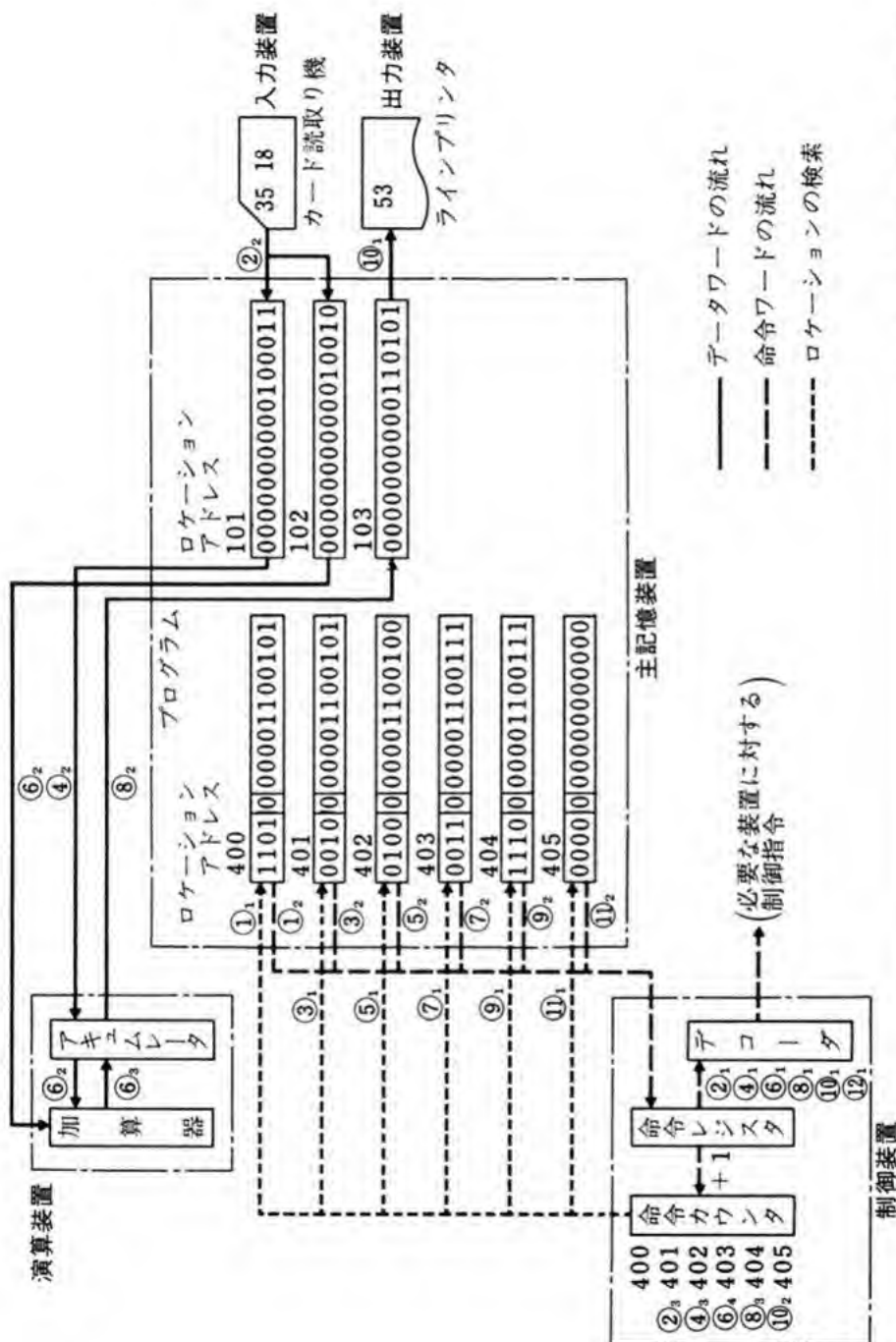



図 15 プログラムの実行とデータの流れ

図 16　*a*, *b* に対応する数値がせん孔されているカード

それを2進数に変換して，主記憶装置のアドレス $(101)_{10}$, $(102)_{10}$ をもつロケーションにストアする。$a+b$ を実行し，その結果 c をアドレス $(103)_{10}$ をもつロケーションにストアする。結果をプリンタで出力する。プログラムは，図17に示すように，アドレス400, 401, …, 405をもつロケーションに順次ストアされている。命令カウンタは，最初400を示している。

スタートキーを押すと，計算機の制御装置では，インストラクションフェッチサイクルが始まる。

① 命令カウンタが400を示しているので，アドレス400をもつロケーションを探して，そこに入っている内容を命令レジスタにロードする。このことによって，命令レジスタに命令ワード |1 1 0 1|0|0 0 0 0 0 1 1 0 0 1 0 1| が収容される。

② 命令レジスタにロードされた命令ワードは，デコーダで解読される。最初の4ビットは1101であるから，入力命令であることが認識され，カード読取り機に読取り動作の指令が出される。同時にまた，ビット4が0であるから，インデックスレジスタを参照せず，かつアドレス部の11ビット |0 0 0 0 0 1 1 0 0 1 0 1| をアドレスレジスタにロードする。このアドレスは，$(101)_{10}$ である。これが主記憶装置に伝えられる。ここで初め

ロケーション アドレス			アドレス部
400	1 1 0 1	0	0 0 0 0 1 1 0 0 1 0 1	$(101)_{10}$
401	0 0 1 0	0	0 0 0 0 1 1 0 0 1 0 1	
402	0 1 0 0	0	0 0 0 0 1 1 0 0 1 1 0	$(102)_{10}$
403	0 0 1 1	0	0 0 0 0 1 1 0 0 1 1 1	$(103)_{10}$
404	1 1 1 0	0	0 0 0 0 1 1 0 0 1 1 1	
405	0 0 0 0	0	0 0 0 0 0 0 0 0 0 0 0	

図 17 機械語で書かれたプログラムの内容

て，カード上にせん孔されている35と18という数値がアドレス $(101)_{10}$, $(102)_{10}$ をもつロケーションに入る動作が起きる。このことによって，アドレス $(101)_{10}$, $(102)_{10}$ をもつロケーションに，それぞれデータワード 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 1 と 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 がストアされる。最後に，命令カウンタの数値に，自動的に1が加えられ，401となって，エクセキューションサイクルが終わる。

③ エクセキューションサイクルが終わったので，再びインストラクションフェッチサイクルが始まる。

命令カウンタは401を示しているので，アドレス401をもつロケーションを探して，そこに入っている内容を命令レジスタにロードする。すなわち，命令レジスタに命令ワード 0 0 1 0 0 | 0 0 0 0 1 1 0 0 1 0 1 が収容される。

④　命令レジスタにロードされた命令ワードは，デコーダで解読され，ロード命令であることから演算装置にロード動作の指令が出され，同時にアドレス部の内容 $(101)_{10}$ が，アドレスレジスタを通って，主記憶装置に伝えられる。ここで初めて，アドレス $(101)_{10}$ をもつロケーションの内容35がアキュムレータにロードされる。最後に，命令カウンタの数値に1が加えられ，402となる。

⑤　命令カウンタは402を示しているので，アドレス402をもつロケーションの内容を命令レジスタにロードする。すなわち，命令レジスタには命令ワード［0 1 0 0|0|0 0 0 0 1 1 0 0 1 1 0］が収容される。

⑥　命令ワードが解読され，加算命令であることから，演算装置に加算動作の指令が出され，同時にアドレス部の内容 $(102)_{10}$ が，アドレスレジスタを通って，主記憶装置に伝えられる。ここで，アキュムレータの内容35に，アドレス $(102)_{10}$ をもつロケーションの内容18が加算され，結果［0 0 0 0 0 0 0 0 0 0 1 1 0 1 0 1］が，アキュムレータに立つ。これは $(53)_{10}$ である。最後に，命令カウンタの数値が403となる。

⑦　命令カウンタが403を示しているので，アドレス403をもつロケーションの内容，すなわち命令ワード［0 0 1 1|0|0 0 0 0 1 1 0 0 1 1 1］が命令レジスタにロードされる。

⑧　命令ワードが解読され，ストア命令であることから，演算装置に加算動作の指令が出され，同時にアドレス部の内容 $(103)_{10}$ が，アドレスレジスタを通って主記憶装置に伝えられる。ここで，アキュムレータの内容53が，主記憶装置のアドレス103をもつロケーションにストアされる。最後に，命令カウンタの数

値が404となる。

⑨　命令カウンタが404を示しているので，アドレス404をもつロケーションの内容 | 1 1 1 0 | 0 | 0 0 0 0 1 1 0 0 1 1 1 | を命令レジスタにロードする。

⑩　命令ワードが出力命令であることから，ラインプリンタに出力動作の指令が出され，同時にアドレス部の内容 $(103)_{10}$ が，アドレスレジスタを通って，主記憶装置に伝えられる。アドレス $(103)_{10}$ のロケーションの内容が10進数に変換された後，ラインプリンタから53という数値が出力される。また，命令カウンタの数値が405となる。

⑪　命令カウンタが405を示しているので，アドレス405をもつロケーションの内容 | 0 0 0 0 | 0 | 0 0 0 0 0 0 0 0 0 0 0 | を命令レジスタにロードする。

⑫　命令ワードが停止命令であることから，停止の指令が出され，計算機が停止する。以上でこのプログラムは終了する。

これらの説明で，奇数番号はインストラクションフェッチサイクルであり，偶数番号はエクセキューションサイクルである。

ここで，記憶装置のアドレス400から405をもつ一連のロケーションに注目しよう。この中に入っている情報は命令である。図17に示すような順序がついた一連の命令の集まりを**プログラム**という。図17のプログラムは，電子計算機がその意味を直接理解できることばで書かれているという意味で，**機械語プログラム** (machine language program)という。ところでわれわれの計算機では，プログラムは，主記憶装置に機械語で記憶されている。このように，計算に必要な数値ばかりでなく，計算手順であるプログラムまで記憶していて，計算処理を行うような計算機を**プログラム記憶式計算機** (stored

program computer）という。

電子式卓上計算器を用いると，計算速度は確かに速い。けれども，能率に限界がある。その原因は，計算処理と計算処理との間に，次に何をなすべきかという手順を決めるのに，人手が介入するからである。言い換えれば，プログラムの進行を人手で行っているのである。プログラム記憶式計算機では，このプログラムの進行をも自動化することによって，計算処理の速度が飛躍的に向上したのである。

問　題

1. IC 記憶装置では，ロケーションをどのようにして選択しているか。
2. 記憶装置の記憶容量は何によって表すか。
3. 電子計算機の周辺装置の主なものを四つ挙げよ。
4. カード・紙テープでは数字や英字をどのようにして表しているか。
5. 磁気テープ装置への記憶方法について説明せよ。
6. 制御装置における命令カウンタ・命令レジスタ・デコーダの働きを説明せよ。
7. 演算装置におけるアキュムレータの働きを説明せよ。

5 ソフトウエア

この節のねらい　電子計算機に処理させようとする仕事は，一連の命令として書き表したプログラムによって実行される。

ここでは，プログラミングに使われる流れ図やプログラム言語などについて学び，ついで簡単な問題について FORTRAN で書かれたプログラムの作り方を調べる。

1. プログラミング

(1) プログラム作成の手順

プログラムを作成するには，解かせる問題や使用する計算機によっても異なるが，およそ図1に示すような手順をとることが多い。そのためには，いろいろな知識が必要である。

問題の分析と仕様作成　電子計算機に行わせる仕事の内容を明確にするために問題の分析を行う。そして，仕事の処理手順を詳細に設定し，仕様書を作成する。

プログラム設計　仕様書を仕事の流れに置き換えた内容として，**流れ図**（フローチャート）を書く。この場合，概略流れ図と詳細流れ図の二つに分けて書くこともある。また，この段階では，この仕事に向いたプログラムを決定しておくことが必要である。

プログラミング　プログラムを作成することを**プログラミング**，または**コーディング**という。プログラムは，流れ図に従って作成する。作成されたプログラムは，カードや端末装置などを使って電子計算機に入力する。

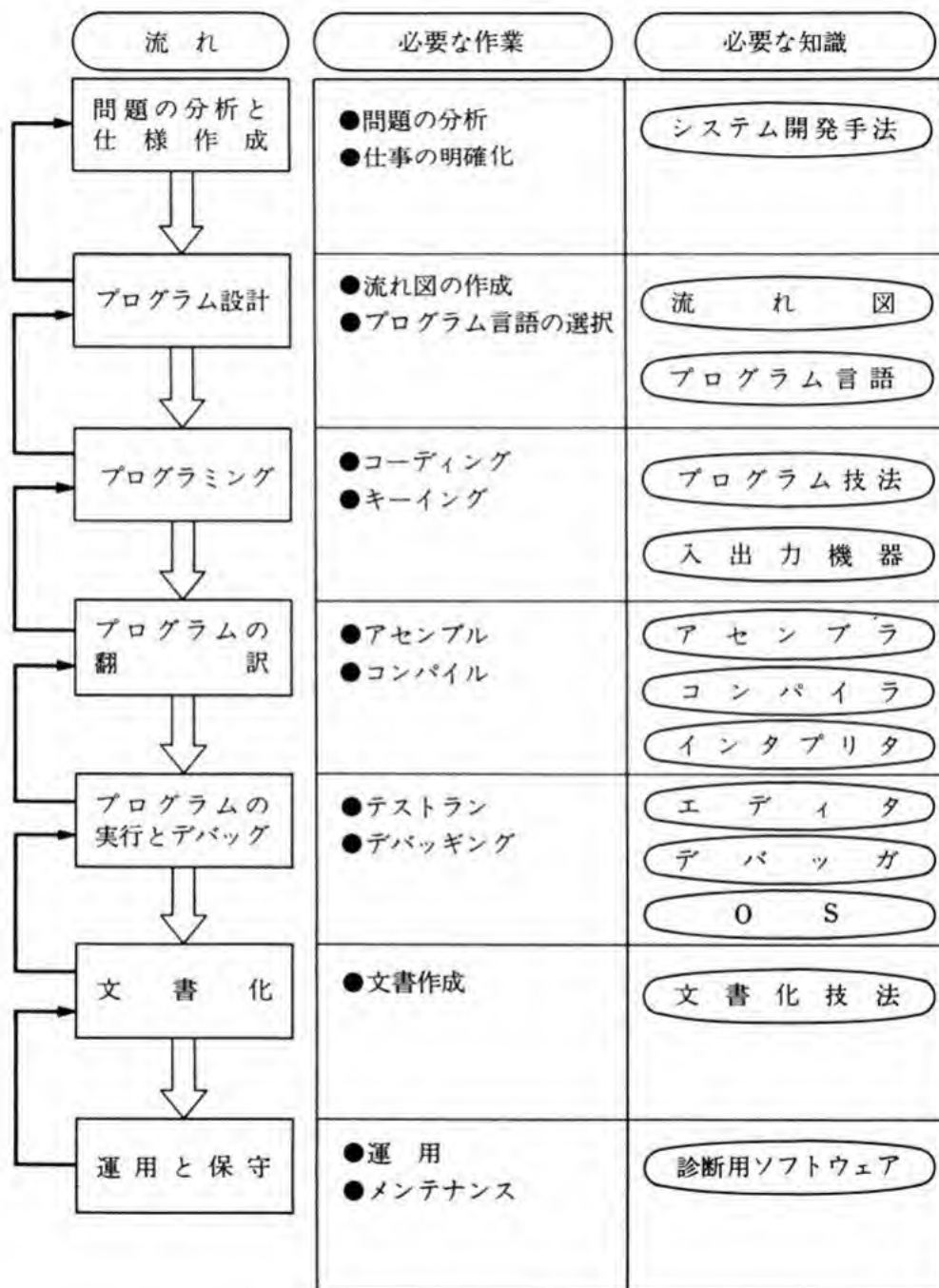
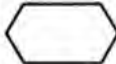
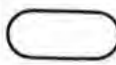



図 1 プログラム作成の手順

プログラムの変換　一般にプログラムは人間にとってわかりやすい高水準言語で書かれているため，機械ではすぐ理解できない。そのため，人間にとって都合のよいプログラム言語を，機械語に直す必要がある。これを**プログラムの変換**という。プログラムの変換は，電子計算機に行わせる。

プログラムの実行とデバッグ　機械語に変換されたプログラムは，直ちにそのプログラムの手順どおりに実行されて，計算結果を

表1　流れ図の記号の例

記号	意味
(長方形)	処　理：あらゆる種類の処理機能を表す。
(ひし形)	判　断：一つの入口といくつかの択一的な出口をもち，記号中に定義された条件の評価に従って，唯一の出口を選ぶ判断機能，またはスイッチ形の機能を表す。
(六角形)	準　備：指標レジスタの変更など，その後の動作に影響を与えるための命令，または命令群の修飾を表す。
(平行四辺形)	データ：媒体を指定しないデータを表す。
(角丸長方形)	端　子：プログラムの流れの開始もしくは終了などの外部環境への出口，または外部環境からの入口を表す。
(波形底の長方形)	書　類：人間が読める媒体上のデータを表す。
(角切り長方形)	カード：カードを媒体とするデータを表す。
(直線)	線　：データまたは制御の流れを表す。
(円)	結合子：同じ流れ図中の他の部分への出口，または他の部分からの入口を表す。また，線を中断し他の場所に続けるのに用いる。

得ることができる。

プログラムが所期のとおり実行できないとき，つまりプログラムに文法上の誤りや考え違いなど論理的な誤りがあったとき，その誤りを見付けて正す必要がある。このプログラムの手直しを**デバッグ**という。デバッグは，プログラムが完全に実行できるまで，何回も繰り返して行う必要がある。

文書化　プログラムが完成した後は，そのプログラムの内容や使い方などを文書にしておくと，後でプログラムを修正するときや，新しいユーザがそのプログラムを使うときに役だつ。

運用と保守　完成したプログラムは，運用と保守を行って評価する。

(2) 流　れ　図

プログラムを作成するとき，その問題を解析して，その作業手順を明確に示すために，まず流れ図を作成する。

流れ図は，電子計算機によって実行される作業の種類・系列，作業の進行などを記号で表す。表1は，JISに定められている流れ図の記号の一部を示したものである。

次の例題を電子計算機によって実行する手順を，流れ図で示すと，図2のようになる。

例題 1.　直径 D をカードから読み取り，円の面積 S を求め，D と対応する S の値をプリンタで印字する。この手順を，流れ図に表せ。

図2 例題の流れ図

解答

ふつう流れ線の流れの向きは矢印で示すが，流れの向きが左から右へ，または上から下への場合は，矢印を省略してもよい。

流れ図を書くことによって，プログラムの作成が容易になり，誤りも少なくなる。また，作成したプログラムを検討するときの助けにもなる。

（3） プログラム言語

流れ図の作成が終わると，プログラミング作業に移るが，そのためにはどのようなプログラム言語を用いるかを決める必要がある。ここでは，プログラム言語の種類・特徴などについて調べる。

機械語　機械語は，p.62 で学んだように，2進数字だけで書かれているので，人間が一般に使っていることばとは，ほど遠いものである。

アセンブラ言語　機械語は，演算コードを初め，ロケーションの内容もすべて2進数字で表しているため，見にくく記憶しにくい。これらを人間にとってわかりやすい記号を用いて，書きやすく記憶しやすくしたものがアセンブラ言語である。

アセンブラ言語で書かれたプログラムの1命令は，原則として，表2のように，機械語の1命令に変換される。アセンブラ言語で書かれたプログラムを，機械語に直すためのプログラムを**アセンブラ**

表 2　四則演算の命令

	機械語	アセンブラ言語
ロード命令	0010	LOD
ストア命令	0011	STO
加算命令	0100	ADD
減算命令	0101	SUB
乗算命令	0110	MPY
除算命令	0111	DIV

表3　機械語・アセンブラ言語・コンパイラ言語の比較

機　械　語	アセンブラ言語	コンパイラ言語
0010000000000100	LOD SURYO	
0110000000000101	MPY TANKA	KINGAK=SURYO * TANKA
0011000000000111	STO KINGAK	

(assembler) という。

コンパイラ言語　コンパイラ言語を用いると，アセンブラ言語を用いるよりも，プログラミングがさらに容易になる。

コンパイラ言語は，比較的人間のことばに近い構成をもつプログラム言語で，表3のように，機械語とはかなり異なっている。表3は，あるものの数量と単価から合計金額を求める場合のプログラムに使われる命令を，各プログラム言語で比較して示したものである。コンパイラ言語で書かれたプログラムの1命令は，機械語の複数個の命令に対応する。コンパイラ言語で書かれたプログラムを，機械語のプログラムに直すプログラムを**コンパイラ** (compiler) という。コンパイラ言語の一つの単位を**ステートメント** (statement，**命令文**) という。

現在，一般に使われているコンパイラ言語としては，図4のように，**FORTRAN** (FORmula TRANslator の略)・**ALGOL**(ALGOrithmic Language の略)・**COBOL**(COmmon Business Oriented Language の略)・**PL/I** (Programming Language one の略) などがある。

言語変換段階と実行段階　FORTRAN などの機械語以外の言語によって書かれたプログラムを**ソースプログラム**という。ソースプログラムを機械語に変換するプログラムを**言語変換プログラム**という。言語変換の結果得られる機械語のプログラムを**オブジェクトプログラム**という。

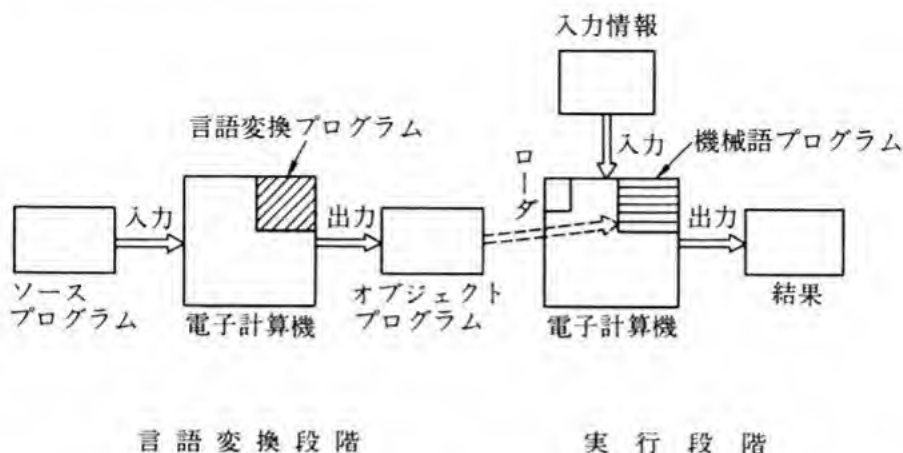



図 3 言語変換段階と実行段階

図3を見てみよう。われわれは，まずソースプログラムを作成する。ソースプログラムを入力し，言語変換プログラムを用いて，オブジェクトプログラムを出力させる。この段階を**言語変換段階**（translation phase）という。

次に，得られたオブジェクトプログラムを**ローダ**（loader）というプログラムを用いて，主記憶装置に読み込ませる。* このことを**ローディング**（loading）という。ここで初めて，必要な入力情報を与えるとプログラムが実行され，結果が得られる。この段階を**実行段階**（execution phase）という。

電子計算機を動作させて必要な結果を得るためには，プログラムを作成して，言語変換段階と実行段階とを経なければならない。つまり，計算機を2回用いることになる。これを**ツーパス**（2 pass）**方式**という。

* p. 57の電子計算機の動作で，あらかじめ，アドレス 400, 401, …, 405 をもつロケーションに順次ストアされているとしたプログラムは，ローダというプログラムを用いて読み込ませたのである。

インタプリティブ言語　コンパイラは書かれたプログラムをすべていちどに機械語に直し，後で実行に移すのに対して，1ステートメントだけ機械語に変換してすぐ実行し，次のステートメントの変換と実行に移る**インタプリタ**（interpreter）がある。インタプリタで変換される言語をインタプリティブ言語（**解釈言語**）といい，BASIC（Beginner's All purpose Symbolic Instruction Code の略）・PASCAL などがある。インタプリティブ言語は，電子計算機と情報をやりとりしながら，プログラムを作成したり，実行したりする会話形言語である。

問題向き言語　問題向き言語には，シミュレーション用・プロセス制御用・数値制御用などが開発されている。

シミュレーションは，システムや現象を模型化し，電子計算機などを用いて実際の状況を模擬することである。シミュレーションさせるために設定したモデルを**シミュレーションモデル**といい，シミュ

図 4　プログラム言語の種類

- プログラム言語
  - はん用プログラム言語
    - 機械語
    - アセンブラ言語
    - コンパイラ言語
      - 科学技術計算用言語　FORTRAN, ALGOL
      - 事務計算用言語　COBOL
      - 多目的用言語　PL/I
      - 会話形言語　BASIC
  - 問題向き言語
    - シミュレーション言語
    - プロセス制御用言語
    - 数値制御用言語

〔例〕

FORTRAN CODING SHEET

ページ 1

業務名		#	作成日	・ ・	検印		作成者	
作業名		#	修正日	・ ・	注記			
プログラム名	電気回路の計算	#		・ ・				

```
C STATEMENT CONT                       FORTRAN STATEMENT                     IDENTIFICATION
1 2     5   6   7  10   15   20   25   30   35   40   45   50   55   60   65   70 72 73 75 80
C               DENKI KAIRO NO KEISAN
                REAL L
                WRITE(6,201)
   201          FORMAT(1H1,27H       OMEGA             Z)
                READ(5,100)R,L,C
   100          FORMAT(2F10.1,F10.7)
     1          READ(5,101)OMEGA
   101          FORMAT(F10.1)
                IF(OMEGA.LE.0.0)STOP
                Z=SQRT(R**2+(OMEGA*L-1.0/(OMEGA*C))**2)
                WRITE(6,200)OMEGA,Z
   200          FORMAT(1H ,2E16.6)
                GO TO 1
                END
```

図 5 コーディングシートへの記入例

レーションを行うシステム・装置またはプログラムを**シュミレータ**という。シミュレーションモデルを書き表すために用いられるプログラム言語が**シミュレーション言語**である。

プロセス制御用言語は，後で学ぶプロセス制御に電子計算機を応用
5 する場合に用いられるプログラム言語である。

数値制御用言語は，後で学ぶ数値制御を行う場合に用いるプログラム言語である。

(4) コーディング

プログラムは，図5のようなコーディングシート(プログラムシー
10 トともいう)に，一定の書式に従って書く。

コーディングシートの各行は80けたからなっており，入力にカードを用いる場合には，1行を1枚のカードにせん孔する。

図5は，後に学ぶ電気回路の計算の流れ図(80ページ図10)をFORTRANでコーディングした記入例である。

15 **問 1.** 科学技術計算用にはどんなプログラム言語が用いられているか。

問 2. アセンブラ・コンパイラとは何か。

問 3. 機械語・アセンブラ言語・コンパイラ言語を比べてその特徴を述べよ。

問 4. 会話形言語とは何か。

2. FORTRANのプログラム例

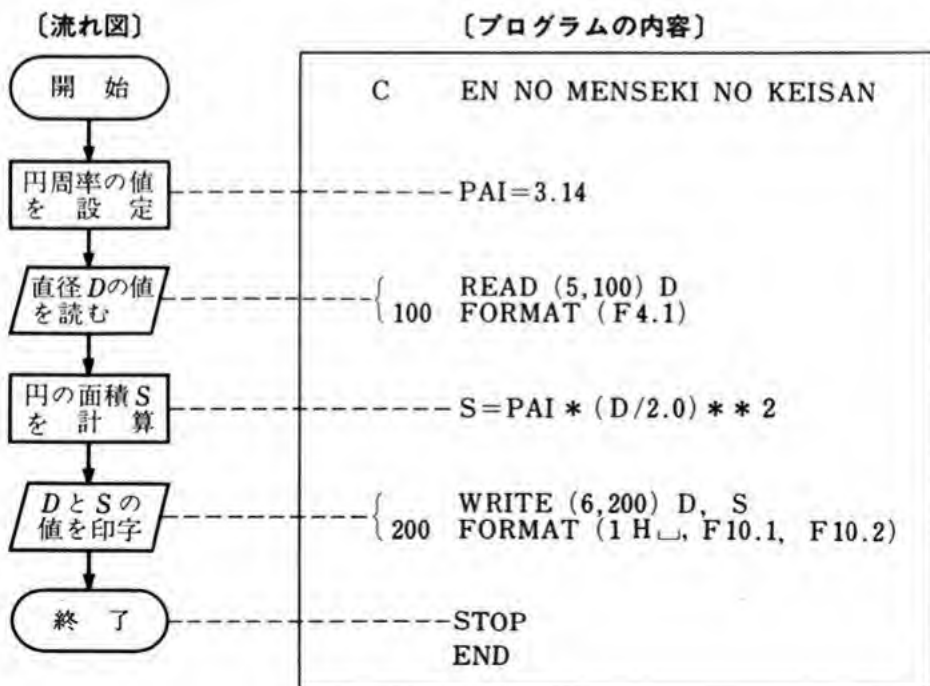
(1) 円の面積の計算

直径 d が与えられたときの円の面積は，$S=\pi\times\left(\frac{d}{2}\right)^2$ である。この計算を電子計算機で行うときのプログラムについて考える。図6は，この場合の流れ図とプログラムを対応させて示したものである。

FORTRAN コーディングシートの1行は，72個の文字（空白も含む）からなっている。

```
C       EN NO MENSEKI NO KEISAN
```

第1カラムにCの文字のある行を**注釈行**という。この行は，プログラムの実行には関係しないので，第2カラム以降には何を書い

図6 円の面積計算の流れ図とプログラムの内容

てもよい。ここでは，プログラムの標題（EN NO MENSEKI NO KEISAN）を示している。

```
PAI=3.14
```

FORTRANでは，命令のことを**文**という。通常は，コーディングシートの第7カラムから第72カラムまでの範囲に，1行に書かれるが，継続行にわたって書かれることもある。継続行には，第6カラムの空白を0以外の文字で埋める。上の文は，PAIに3.14という値を与えることを意味している。ここで，πの代わりにPAIとしたのは，FORTRANで使用できる文字には，表4のような制限があるためである。

なお，プログラムをコーディングシートに記入する場合，キーパンチャが文字を読み誤らないように注意する必要がある。例えば，0, O, Dを区別するために，Oを$\overline{\text{O}}$あるいはØ，DをĐのように書くことに決めておくとよい。また，1とIを区別するために，Iの上下の横線をはっきりと書くなどの注意も必要である。

FORTRANでは，データワードのアドレスを数字ではなく，記

表 4 FORTRANで使用できる文字

種類	文字（文字の名称）
英字（26字）	A, B, C, ……, Z
数字（10字）	0, 1, 2, ……, 9
特殊文字（11字）	（空白） = （等号） + （プラス） − （マイナス） * （星印） ／ （斜線） (（左括弧）) （右括弧） , （コンマ） . （点） ' （アポストロフィ）

注. JISでは，FORTRANの機能によって2種類に分け，基本水準と，より多くの機能を備えた上位水準とがある。この教科書では，基本水準に基づいてプログラムを説明する。なお，表中の（空白）は，文字の書かれていない部分であるが，以後の説明では，便宜上，␣を用いて示す。

号で，例えばPAIのように名前をつけて表す。これを**変数名**という。PAI=3.14としておけば，以後PAIと書くだけで円周率を表すことになる。なお，3.14のように，小数点のついた実数型の定数を**実定数**といい，小数点のついていない整数型の定数を**整定数**という（表5）。

表5 整定数と実定数

整定数	2 6 100
実定数	2.0 3.14 10.3

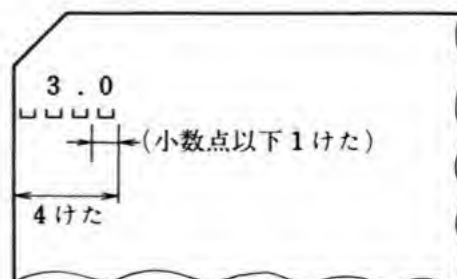
変数名は，英字で始まる6文字以内の英字と数字のつながりからなっている。変数名にも整数の値をとる整数型と実数の値をとる実数型がある（表6）。**整数型の変数名**は，先頭の英字がI, J, K, L, M, Nのうちのどれかで表し，**実数型の変数名**は，それ以外の英字で始まると決めてある。それ以外の変数名の使い方をする場合には，後で学ぶ型宣言文を用いる必要がある。

```
READ (5 , 100) D
```

この文は，5という番号のついた入力装置（一般にはカード読取り機）から，データを読み取って変数名Dの値にする。つまり，Dという名前のついた主記憶装置のロケーションにストアすることを意

表6 変数名の書き方

	正しい例	誤りの例
整数型の変数名	I J LENGTH	3I A TIME
実数型の変数名	X Y RATE	DISTANCE A−B MASS

図7 F4.1によるデータの形式

味し，これを **READ 文**という。文中の（　）内の100は，読み取るデータの形式を指定しているFORMAT文の番号を示す。なお，この文のように，直径に対してDという変数名で表される変数を用いるのは，いろいろな値の直径に対して計算できるプログラムにするためである。この場合は，直径dの値をデータカードにせん孔して，それを読み取る。

```
100 FORMAT (F 4.1)
```

この文は，前のREAD文中で指定されているデータの読取り形式を示したもので，**FORMAT 文**という。

上のFORMAT文で100を**文番号**といい，文番号はコーディングシートの第1カラムから第5カラムの範囲に書く。また，F4.1を**編集記述子**といい，F4.1は，読み取るデータが図7のように，小数点も含めて4けた以内で，その値が小数点以下1けたの実数型の変数であることを示している。

```
S=PAI*(D/2.0)**2
```

この文は，面積を計算するものである。* は乗算，/ は除算，** はべき乗（累乗）を表す記号で，これらを**算術演算子**という（表7）。等号の右辺は**算術式**，左辺と右辺を含めて**算術代入文**（図8）という。

等号「＝」は，右辺の算術式の結果を左辺の変数名のロケーションにストアすることを意味し，数学で扱う等号の意味とは異なる。等号の左辺は変数名でなければいけない。

表 7　算術演算子

記　号	意　味
＋	加　算
－	減　算
*	乗　算
／	除　算
**	べき乗

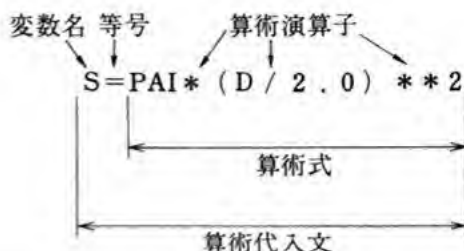



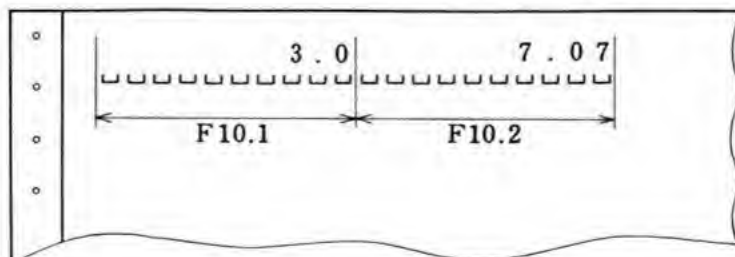
図 8　算術式と算術代入文

```
WRITE (6 , 200) D , S
```

この文は，6という番号のついた出力装置（一般にはラインプリンタ）によって，計算結果として，DとSの値を印字することを意味し，これを**WRITE 文**という。また，文中の（　）内の200は印字の形式を指定する文番号である。

```
200  FORMAT ( 1 H␣, F 10.1 , F 10.2)
```

この文は，前の WRITE 文中で指定した印字形式を示したものである。編集記述子の順序は，WRITE 文の変数の書かれた順序と対応する。F 10.1 は変数 D，F 10.2 は変数 S の形式を指定している。編集記述子の先頭の 1 H␣ は，次の行から印字することを表しているので，改行後，図9のように印字される。図9は，直径 d＝3.0 の場合を示す。

図 9　F 10.1，F 10.2 による印字の形式

```
STOP
```

この文は，FORTRAN プログラムの実行を終了させるためのもので，**STOP 文**という。

なお，FORTRAN プログラムの最後の行には，次の **END 文**を

5 書くことになっている。

```
END
```

問 5. 算術演算子について述べよ。

(2) 電気回路の計算

RLC 直列回路のインピーダンス Z は，$Z=\sqrt{R^2+\left(\omega L-\frac{1}{\omega C}\right)^2}$

10 から求められる。ここでは R, L, C の値をカードから読み込んで，ω のいろいろな値に対する Z の値を求める場合のプログラムについて考える。図 10 は，この場合の流れ図とプログラムを対応させて示したものである。

このプログラムで使用する変数は，すべて実数型である。しかし，

15 前に学んだように，L は整数型の変数であるので，次のような文によって，L が実数型の変数であることを宣言しておく。

```
REAL L
```

この文を**型宣言文**という。また，整数型の変数であることを宣言するときは，INTEGER と書く。

20 印字用紙は，送りがしやすいように，各ページが連続した用紙が使われる。ページの先頭から印字するには，FORMAT 文のかっこ

〔流れ図〕

〔プログラムの内容〕

```
      REAL L

      WRITE(6, 201)
  201 FORMAT(1H1, 27H␣␣␣␣␣␣␣␣␣␣OMEGA␣␣␣␣␣␣␣␣␣␣␣Z)

      READ (5, 100) R, L, C
  100 FORMAT ( 2 F10.1,  F10.7)

    1 READ (5, 101) OMEGA
  101 FORMAT (F10.1)

      IF (OMEGA.LE.0.0) STOP

      Z=SQRT (R**2+(OMEGA*L-1.0/(OMEGA*C))**2)

      WRITE (6,200) OMEGA, Z
  200 FORMAT ( 1H␣,  2E16.6)
      GO TO 1

      END
```

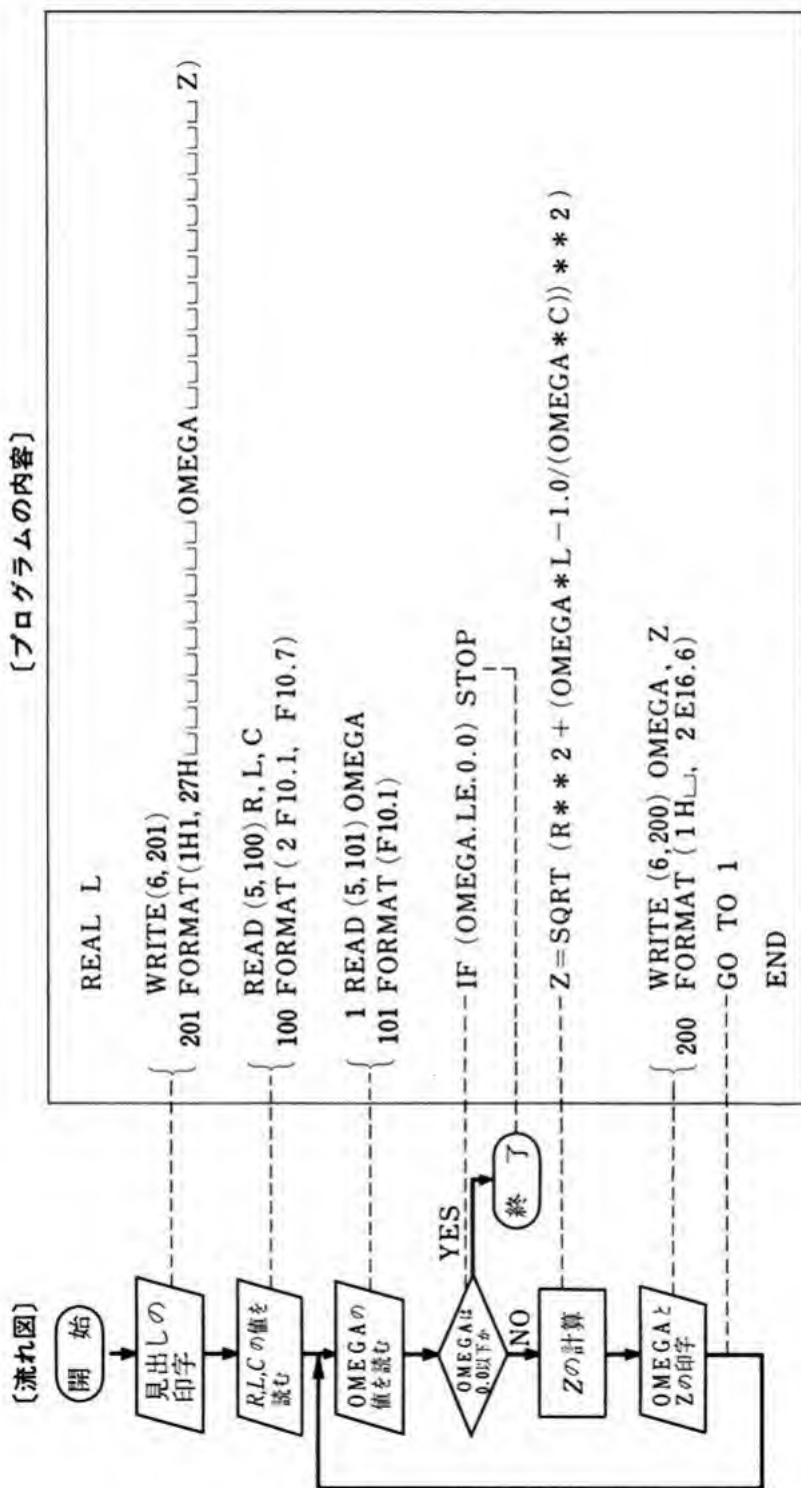



図 10　*RLC* 直列回路の計算の流れ図とプログラムの内容

の最初に，次のように書けばよいことになっている。

```
1 H 1
```

なお，いままで印字した次の行に印字したい場合には1H␣，1行あけた次の行に印字したい場合には1H0と書けばよい。

WRITE文に伴うFORMAT文は，必ず1H1，1H␣，または1H0で始めなければならない。

計算結果を見やすくするための見出しを印字するには，FORMAT文の中に次のように書けばよい。

```
27H␣␣␣␣␣␣␣␣␣OMEGA␣␣␣␣␣␣␣␣␣␣␣␣Z
```

FORMAT文の中にこのように書くと，Hという文字の次の27個の文字が，そのまま印字される。改ページと見出しの印字を，WRITE文とFORMAT文によって行う例を図11に示す。

変数 R, L, C の値をカード読取り機から文の番号100のFORMAT文に従って読み込ませるには，次のように書く。

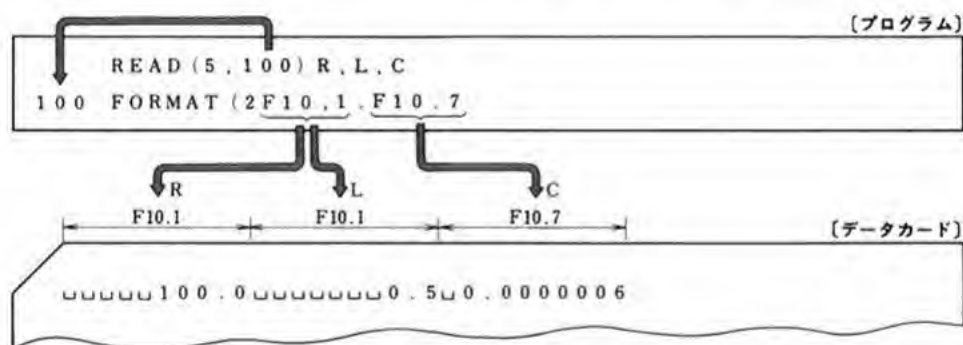
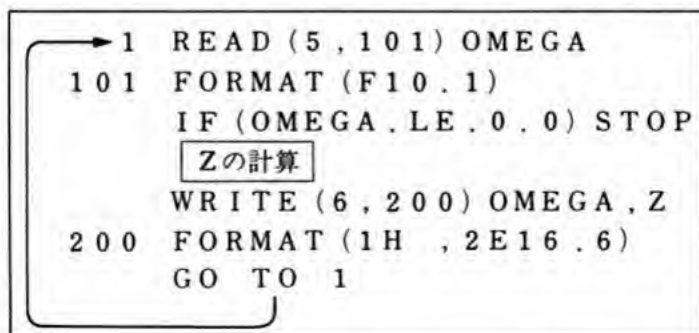
図 11　WRITE文，FORMAT文と印字結果との対応

```
      READ (5 , 100) R , L , C
100   FORMAT (2F10.1 , F10.7)
```

2F10.1, F10.7は，F10.1が2個，F10.7が1個，すなわち2個の変数R, LがそれぞれF10.1，変数CがF10.7の形式で読み取られることを示す。図12は，R, L, Cの値がそれぞれ100.0〔Ω〕，0.5〔H〕，0.0000006〔F〕の場合を示す。しかし，データカードにせん孔された形式と，FORMAT文の編集記述子の形式とが一致しない場合は，データカードにせん孔されたデータの形式に従って，読み取られる。

図12 READ文，FORMAT文とデータとの対応

図13 データの繰り返し読み取り

```
  1  READ (5,101) OMEGA
101  FORMAT (F10.1)
     IF (OMEGA.LE.0.0) STOP
     [Zの計算]
     WRITE (6,200) OMEGA,Z
200  FORMAT (1H ,2E16.6)
     GO TO 1
```

なお，インピーダンス Z は，OMEGA を 500.0, 1000.0, ……, 5000.0 のように与え，繰り返し計算したいので，図 13 のように，

```
GO TO 1
```

と書いた **GO TO 文**を使う。GO TO 1 は，文の番号 1 のついた文，すなわちここでは，

```
1  READ (5 , 101) OMEGA
```

へ行けという意味である。

データの最後は 5000.0 であるが，データの最後を 5000.0 の後に 0.0 としておけば，次の論理 IF 文で，繰り返し読み取ることを停止させることができる。

```
IF (OMEGA . LE . 0.0) STOP
```

この文は，もし（　）内の式がなりたてば，STOP 文を実行して停止，そうでなければ，次の文へ進むことを示している。（　）内の .LE. は，≦ の記号の意味を表し，次の式と等価である。

$$\text{OMEGA} \leqq 0.0$$

論理 IF 文のかっこ内は **論理式** とよばれ，この式がなりたつとき，これを「真」であるといい，なりたたないときは「偽」であるという。したがって，論理 IF 文は図 14 のような働きをする。

図 14　論理 IF 文

表 8　論理式を表す演算子

関係演算子

関係演算子	意　味	関係演算子	意　味
.LT.	$<$	.NE.	$\neq$
.LE.	$\leqq$	.GT.	$>$
.EQ.	$=$	.GE.	$\geqq$

論理演算子

論理演算子	意　味
.OR.	論理和
.AND.	論理積
.NOT.	否　定

表 9　FORTRAN で使用できる関数の例

関　数	書き方	意　味	関　数	書き方	意　味
平方根	SQRT(X)	$\sqrt{x}$	絶対値	ABS(X)	$\|x\|$
指　数	EXP(X)	e^x	実数化	FLOAT(N)	整数型を実数型に変換する
自然対数	ALOG(X)	$\log_e x$			
正　弦	SIN(X)	$\sin x$			
余　弦	COS(X)	$\cos x$			

実行文 は，プログラムの動作を指定するもので，FORTRAN の文には，このほかに非実行文がある。**非実行文** は，プログラムの仕様，データの特性などについて述べるものである。

論理式のところには，表8のような **関係演算子** で結ばれた式か，**論理演算子** で結ばれた式が書かれる。

インピーダンス Z を求める算術代入文を，FORTRAN の約束に従って書くと，次のようになる。

```
Z=SQRT(R**2+(OMEGA*L-1.0/(OMEGA*C))**2)
```

平方根の $\sqrt{\ }$ の記号は SQRT で表し，このように書いただけで，平方根が計算される。このほかに，各種の関数が簡単に書けるようになっている。その一部を表9に示す。

インピーダンス Z を計算した後，OMEGA と Z を対応させて印字するには，次のように書く。

```
     WRITE (6 , 200) OMEGA , Z
200  FORMAT (1 H␣, 2 E 16.6)
```

FORMAT文のE 16.6は，図15のように，データの形式が指数つきの16けたで，小数点以下6けたまでをとることを表している。
5 すなわち，次の形になる。

␣␣␣␣0．××××××E±××

ここで，×は数値が入ることを表している。また，−500.0というような負の数値をE 16.6で表すと，数値の前のけたに−が現れ，次のようになる。

10 ␣␣␣−0．500000E+03

6けた

16けた

なお，指数部の符号が+のときに，+が出力されず空白となる計算機もある。

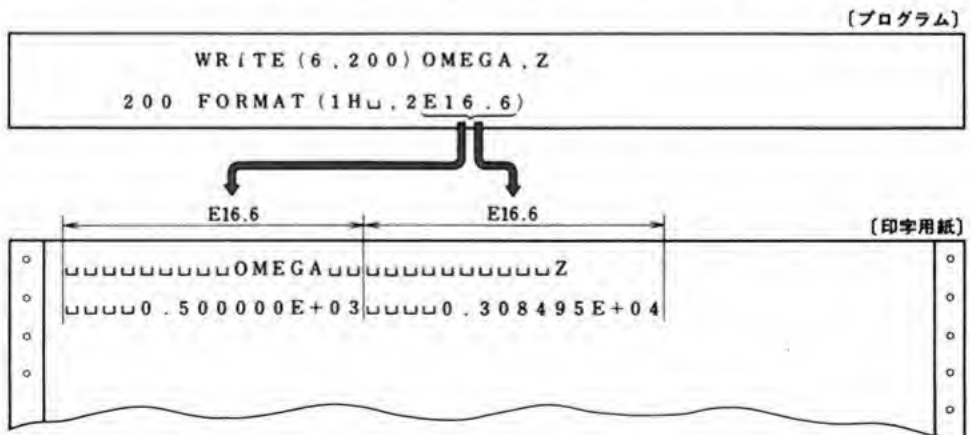
図 15 WRITE文，FORMAT文と印字結果との対応

```
␣␣␣␣␣␣␣␣␣OMEGA␣␣␣␣␣␣␣␣␣␣␣␣Z
␣␣␣␣0.500000E+03␣␣␣␣0.308495E+04
␣␣␣␣0.100000E+04␣␣␣␣0.117094E+04
␣␣␣␣0.200000E+04␣␣␣␣0.194365E+03
␣␣␣␣0.300000E+04␣␣␣␣0.949724E+03
␣␣␣␣0.400000E+04␣␣␣␣0.158649E+04
␣␣␣␣0.500000E+04␣␣␣␣0.216897E+04
```

図16 計算結果

すなわち，-0.5×10^3という書き方に相当し，Eという記号が指数の底10に，Eの次の+03が3乗に相当している。この表し方を**浮動小数点表示**という。これに対して−500.0で表しているものを**固定小数点表示**という。このプログラムの場合は，2E16.6になっているので，OMEGAとZの両方とも，E16.6の形式で印字されることになる。

WRITE文，FORMAT文と印字結果との対応は，図15のようになる。

これまでのプログラムにEND文をつけ加えると，図10に示したFORTRANプログラムになる。また，計算結果を図16に示す。

問 6. 型宣言文について述べよ。

(3) 並べ換え

大小さまざまの数を小さい順に並べ換えるプログラムについて考える。ここでは，表10のような3個の変数X_1, X_2, X_3にデータが入っており，これを並べ換えて$X_1\leqq X_2\leqq X_3$とすることを考えることにする。図17は，この場合の流れ図を，また図18はプログラムを示したものである。

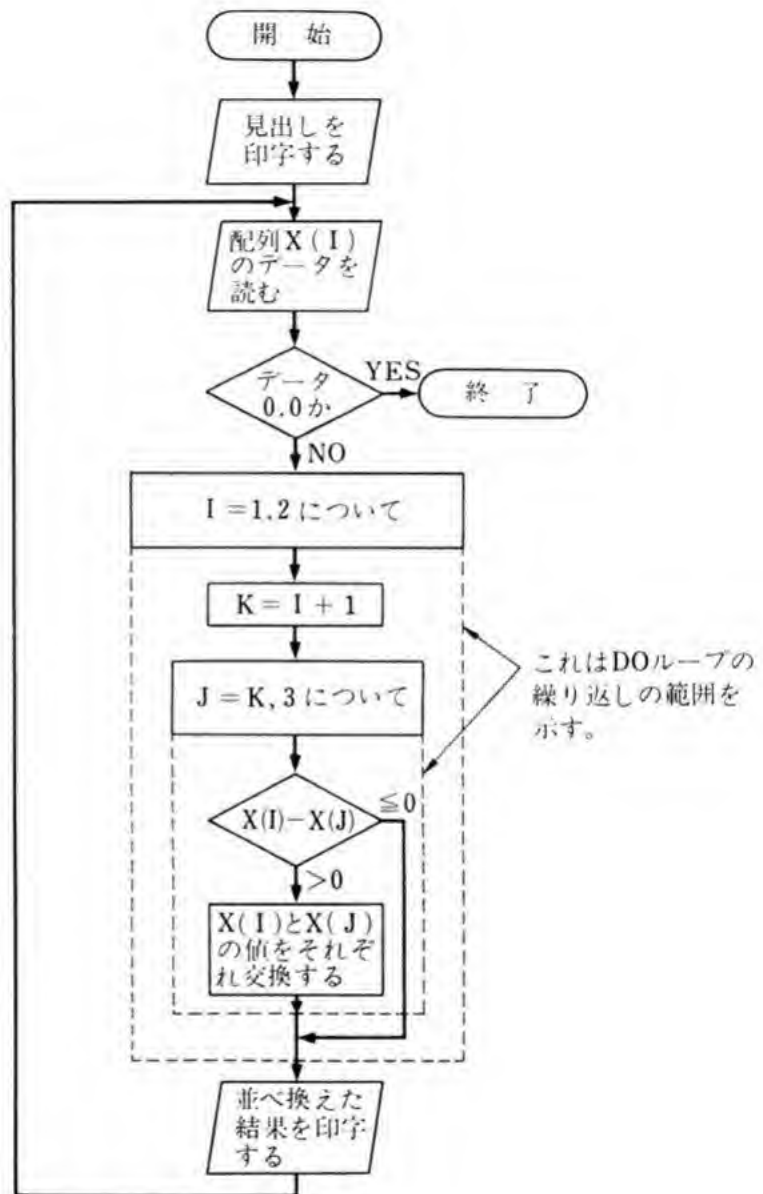



図 17 並べ換えのプログラムの流れ図

表 10 並べ換えようとするデータの例

	X_1	X_2	X_3
第1の組	7.0	5.0	3.0
第2の組	4.0	5.0	1.0
第3の組	9.0	2.0	4.0
第4の組	6.0	9.0	2.0
第5の組	7.0	3.0	8.0
第6の組	0.0	0.0	0.0

（例）
第1の組の数を小さい順に並べ換えると次のようになる。

3.0 5.0 7.0

図 18 並べ換えのプログラム

FORTRAN CODING SHEET

業務名　　#　　作成日　　検印
作業名　　#　　修正日　　注記
プログラム名　並べ換え　　#

```
C     NARABEKAE
      DIMENSION X(3)
      WRITE(6,200)
  200 FORMAT(1H1,24H   X(1)   X(2)   X(3))
 1000 READ(5,100) (X(I),I=1,3)
  100 FORMAT(3F8.1)
      IF(X(1).EQ.0.0)STOP
      DO 10 I=1,2
      K=I+1
      DO 2 J=K,3
      IF(X(I)-X(J)) 2,2,1
    1 T=X(I)
      X(I)=X(J)
      X(J)=T
    2 CONTINUE
   10 CONTINUE
      WRITE(6,201) (X(I),I=1,3)
  201 FORMAT(1H0,3F8.1)
      GO TO 1000
      END
```

変数 X_1, X_2, X_3 については, FORTRAN では X(I) の形式で表され, これを**配列**といい, X は**配列名**という。また, I は, 整定数, 整数型の変数であって**添字式*** とよばれる。したがって X_1, X_2, X_3 は X(1), X(2), X(3) で表し, これらを**配列要素**という。配列を使うと, 一つの配列名で多くの数をひとまとめにして表すことができ, プログラムが非常に簡単になる。

以下このプログラムの要点を説明する。

```
DIMENSION  X(3)
```

これは, 配列が X(1), X(2), X(3) の3個からなることを定義した宣言文で, **DIMENSION 文**という。

```
      WRITE (6 , 200)
  200 FORMAT (1H1 , 24H␣␣␣␣␣X(1)␣␣␣␣␣X(2)
                                 ␣␣␣␣␣X(3))
```

(紙面の都合で2行になっているが, これは1行に書かなければいけない。)

これは, 見出しを印字するための文で, この場合のように WRITE 文に変数を書かないで, 見出しの印字だけに使うことも多い。

```
1000  READ (5 , 100)  (X(I) , I=1 , 3)
```

この文は, X(I) の I の値を最初 I=1 にして X(1) を読み取り, 次に, I に1を加えて I=2 として X(2) を読み, さらに I に1を加えて I=3 として X(3) を読み取ることを意味している。これを

* 添字式には, このほか, 5*J+2, 8*I, J+3 などのような算術式を書くこともある。

DO 形並びといい，I=1,3 は，I を1から始めて1ずつ加え，3になるまで繰り返すことを表す。上の READ 文は，次の文と同じことを意味する。

1000 READ (5, 100) X(1), X(2), X(3)

IF (X(I) . EQ . 0.0) STOP

この文は，前に学んだように，論理 IF 文であって，X(I) が0.0であれば，( ) 内の論理式が真となって STOP 文が実行され，計算が停止される。

DO 10 I=1, 2

この文は，文番号10までのプログラムについて，I=1 の場合を実行し，続いて I=2 の場合を実行することを表し，これを **DO 文**という。I=1, 2 は，I を1から始めて，1ずつ加え，2になるまで繰り返すことを意味している。

DO 文の形式と文の意味

DO n $i=m_1, m_2, m_3$

n: DO 文に対応する端末文の番号 i: DO 変数

m_1: 初期値パラメタ m_2: 終値パラメタ m_3: 増分パラメタ

(文の意味)

最初は $i=m_1$ とし，DO 文の次の文から文の番号 n までの計算を行い，次に i に m_3 を加えて $i=m_1+m_3$ とし，i_n が m_2 を超えない限り，i に m_3 を順に加えて同じ計算を繰り返す。

なお，文の番号 n のついた文は**端末文**という。これは，一つの DO

文と対になっており，DO 文の後になければならない。端末文は一つの DO 文の終わりを示し，CONTINUE 文が使われることが多い。また，m_3 を省略した場合は $m_3=1$ とみなされる。

```
K=I+1
```

5 これは，X(1)とX(2)，X(1)とX(3)，X(2)とX(3)を比較させるための準備として，I=1のときはK=2，I=2のときはK=3とすることを表している。

```
DO 2 J=K, 3
```

これは，文の番号2までのプログラムをJ=K，すなわち，J=2
10 から始めて，3まで繰り返すことを表している。

```
IF (X(I)-X(J)) 2, 2, 1
```

これは，X(I)−X(J)の値が負または0のときは，文の番号2を実行し，正のときは，文の番号1を実行することを表している。これを**算術 IF 文**という。すなわち，ここではX(1)>X(2)，
15 X(1)>X(3)，X(2)>X(3)のとき，文の番号1から2までを実行する。そして，小さい値の順に，X(1)，X(2)，X(3)となるように入れ換え（並べ換え）を行う。

算術 IF 文の形式と文の意味

IF (□) k_1, k_2, k_3

20 □：算術式　k_1, k_2, k_3：算術式の評価に対応する実行文の番号

（文の意味）

□の算術式の計算の結果によって，

負のときは k_1 の文
0のときは k_2 の文
正のときは k_3 の文
を実行する。

T＝X（I）

これは，TにX（I）の値を入れることを表す。

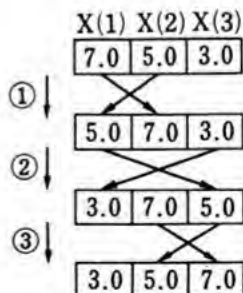
X（I）＝X（J）

これは，X（I）にX（J）の値を入れることを表す。

X(J)＝T

これは，X（J）にTの値を入れることを表す。X（I）とX（J）の値を入れ換える場合，Tを媒介として使わなければならないのは，例えば，X（I）＝X（J）を実行すると，X（I）がX（J）の値となり，X（I）の値が消えてしまうので，X（I）の値を一時Tに移しておく必要があるからである。

図 19 並べ換えの手順（第1の組の結果）

	I=1	I=2
J=2	① X(1)>X(2) ならば X(1) と X(2) を交換	
J=3	② X(1)>X(3) ならば X(1) と X(3) を交換	③ X(2)>X(3) ならば X(2) と X(3) を交換

これまでの並べ換えの手順を整理すると，図19のようになる。

```
2 CONTINUE
```

この文は，「DO 2 J=K, 3」の繰り返し範囲の最後を示すもので，**CONTINUE 文**という。したがって，第1回目が終わってこの文までくると，再び「DO 2 J=K , 3」にもどって繰り返しが始まる。

```
     WRITE (6 , 201)  (X ( I ) ,I=1,3)
201  FORMAT (1H0 , 3F8.1)
```

これは前に学んだ READ 文と同じ DO 形並びであり，次の文と同じ意味である。

```
     WRITE (6 , 201) X ( 1 ) , X ( 2 ) , X ( 3 )
201  FORMAT (1H0 , 3F8.1)
```

```
GO TO 1000
```

これは，一つの組の並べ換えが終わって，結果を印字したところで，次の組の並べ換えを行うため，文番号1000にもどることを表す。

問 7. 次の変数名は整数型か実数型か。

(1) DENKI (2) KEISU (3) NO3 (4) CORD

問 8. 次の式を算術式で表せ。

(1) ax^2+bx+c (2) $\dfrac{b+c}{2a^2}$ (3) $\dfrac{a}{(b+c)^2}$

(4) $\sqrt{a^2-r^2}$

問　　題

1. 次のプログラム言語は，どんな目的に使われているか簡単に述べよ。
   (1) FORTRAN　(2) BASIC　(3) PL/I　(4) ALGOL
   (5) COBOL
2. FORTRANで使用できる文字を挙げよ。　5
3. 図6（p. 74）のプログラムで，次の用語はどの部分をさすか。
   (1) 文　(2) READ文　(3) WRITE文　(4) FORMAT文
   (5) 変数名　(6) 等　号　(7) 算術演算子　(8) 算術式
   (9) 算術代入文　(10) 編集記述子　(11) 整定数　(12) 実定数
4. 論理IF文・算術IF文の働きについて説明せよ。　10
5. 次の用語について説明せよ。
   (1) 浮動小数点表示　(2) 固定小数点表示
6. FORTRANコーディングシート記入の約束について述べよ。

アナログ電子計算機

この節のねらい　アナログ電子計算機は，電圧・電流といった物理量の連続的変化によって動作するものであり，微分方程式を解くとか，ある物理系の動作を調べるのに適している電子計算機である。

ここでは，アナログ電子計算機の構成や，そこで使われる各種の演算器などのしくみについて調べる。

1. アナログ電子計算機のあらまし

(1) 微分方程式とアナログ電子計算機

自然現象や社会現象などの問題には，その解決に微分方程式を利用する場合が多い。微分方程式は，$\dfrac{dy}{dt}$ という微分記号を含んだ方程式である。例えば，列車の速度を求めるには，列車の走行距離を走行時間で割れば求められる。この計算方法では，時速・分速・秒速を求めることができる。秒速よりさらに細かな時間における速度を求めるには，$\dfrac{dy}{dt}$ という記号を使用して，その微小時間における速度を求める。このような微分方程式を解くのに適している電子計算機が，図1のアナログ電子計算機である。次章で学ぶフィードバック制御系の問題にも微分方程式を含むものが多くあり，これらの問題の解決にも利用される。

(2) アナログ電子計算機の構成

アナログ電子計算機は，計算を行う演算部，演算部内にあるいろいろな演算器を連結させる演算器接続部，結果を示す解記録部および電源部などから構成されている。

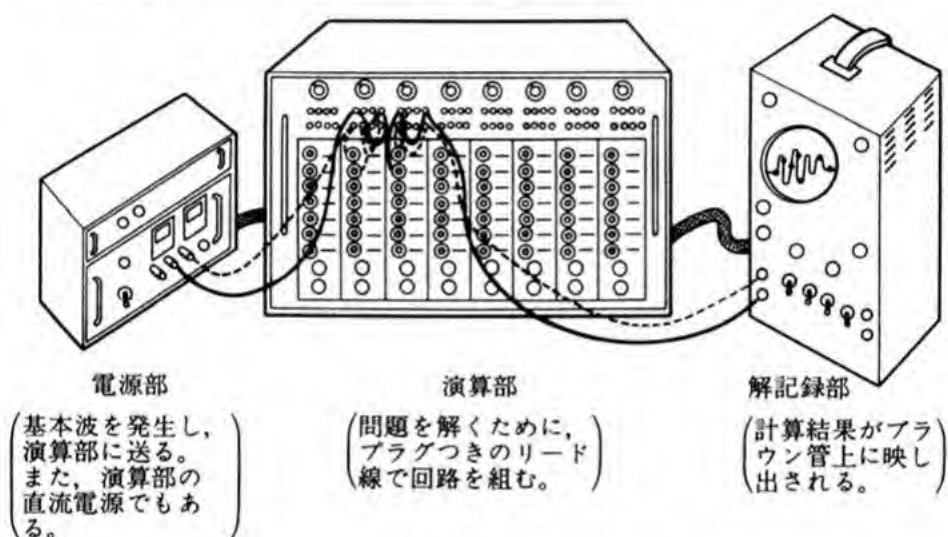
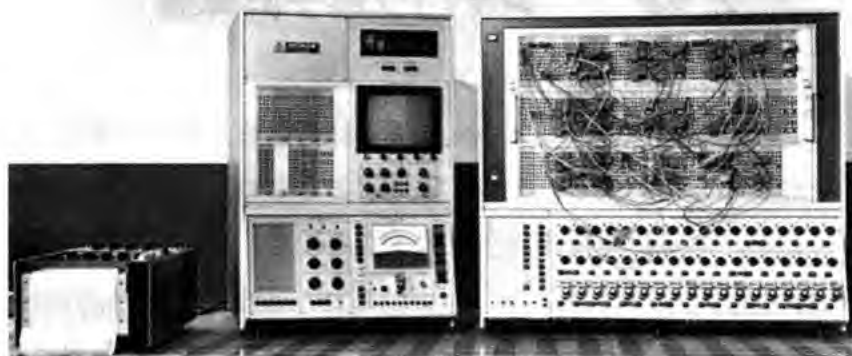



図1　アナログ電子計算機（卓上形）

演算部　アナログ電子計算機の最も中心となるところであり，図2のようないろいろな**演算器**からできている。これらの演算器を工夫して接続することにより，いろいろな計算を行うことができる。

演算器接続部　いろいろの演算器を相互に接続することを容易にするため，すべての演算器の入力・出力端子を1か所に集めた部分である。

解記録部　アナログ電子計算機では，計算の結果をグラフで表すので，そのために便利なブラウン管オシログラフやペンオシログラフおよびX-Yレコーダなどが用いられる。

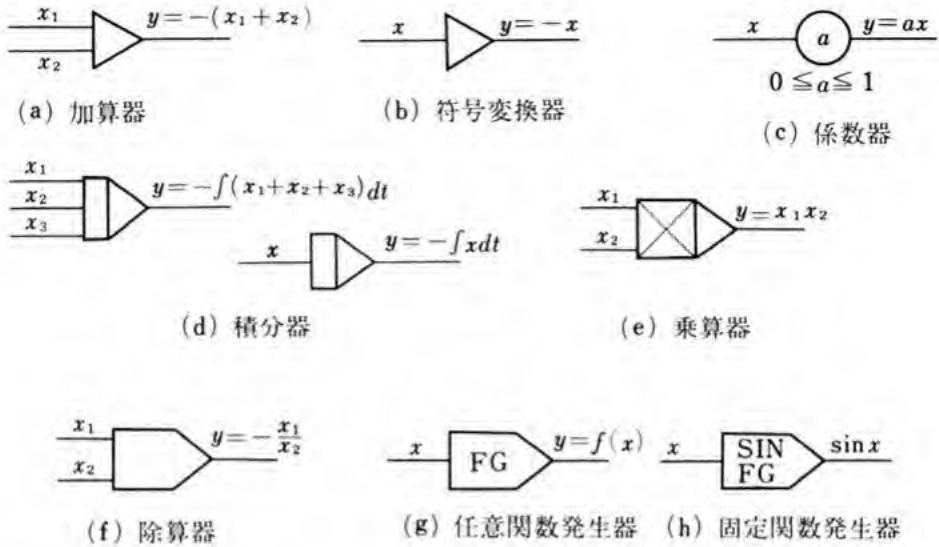



図 2 演算器と図記号の例

(3) 演 算 の 原 理

図3(a)の RL 回路でスイッチSを閉じたとき，回路に流れる電流を i とすれば，キルヒホッフの法則から次の微分方程式がなりたち，その解は図(b)で表される。

$$L\frac{di}{dt}+Ri=V \qquad (1)$$

この微分方程式をアナログ電子計算機を用いて解く場合を考えてみよう。

式（1）は次のように変形できる。

図 3 RL 直列回路の電流

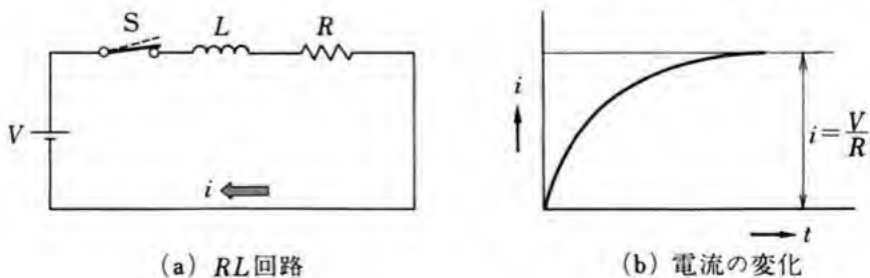
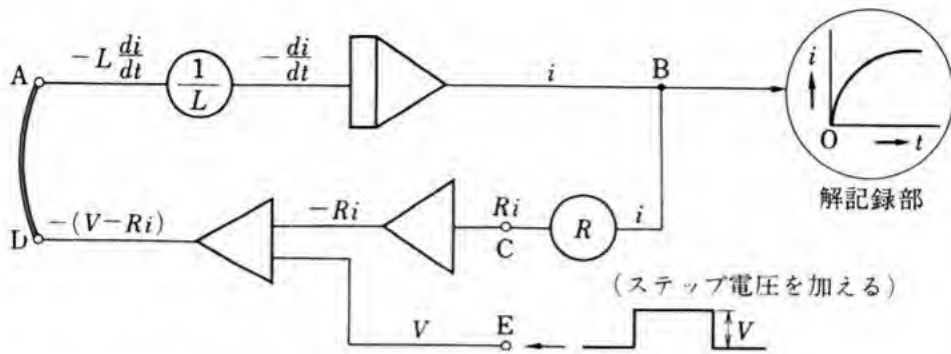



図 4　アナログ電子計算機による演算の原理

$$L\frac{di}{dt}=V-Ri \qquad (2)$$

演算器の入力・出力はすべて電圧であるから，図4のように，A点が$L\frac{di}{dt}$に相当する電圧とすれば，B点はiに相当する電圧となり，D点は$V-Ri$に相当す電圧となる。そこで，A点とD点とを導線で結べば，両点の電位を等しくすることができ，式(2)と同じような関係を作ることができる。

したがって，B点を解記録部に接続すれば，電流iの解が得られることになる。この場合，電流iは，アナログ電子計算機ではB点の電圧となって得られるので，あらかじめ，1〔A〕がアナログ電子計算機では，何ボルトに相当するかを定めておかなければならない。

(4) 換算係数

アナログ電子計算機で方程式を解く場合，変数や数値は電圧に置き換えて計算する。したがって，変数や数値は，アナログ電子計算機では何ボルトに対応するかを決めておく必要がある。例えば，変数qを電圧vに換算するのに，

$$v=\alpha q \qquad (3)$$

として，1種の比例変換を行うとき，このαを**換算係数**という。

アナログ電子計算機の各演算器は，取り扱える最大電圧が規定されているので（ふつう ±50〔V〕程度），これを超えないようにαを決める。

時間についても同じことがいえる。実際にはt〔s〕間の現象を，アナログ電子計算機に最適なτ〔s〕間に行わせるためには，時間の変換が必要になる。このとき，

$$\tau=\alpha_t t \qquad (4)$$

のα_tを**時間換算係数**という。

(5) アナログ電子計算機の種類

アナログ電子計算機には，低速度形と高速度形（繰り返し形）とがある。

低速度形は，1回の演算時間が数秒から数十秒で，解記録部には，ペンオシログラフやX-Yレコーダが使われる。

高速度形は，同じ計算を1秒間に数十回程度の一定周期で繰り返し，解記録部にはブラウン管オシログラフが使われる。したがって，高速度形の場合は，ブラウン管上を直視しながら，定数を変えたときの結果の変化の状態を直ちに知ることができるので便利であるが，低速度形に比べて精度が劣る。

低速度形は，演算の時間がいろいろ変えられるので，実際に起こる現象と同じ時間で解をえがかせることができるので便利である。このような演算のしかたを**実時間演算**という。高速度形は，一定の周期で演算を繰り返すので，実時間演算には適さない。

(6) アナログ電子計算機の応用

アナログ電子計算機では，一次方程式・連立方程式などの代数方程式や微分方程式などを解くことができる。

また，自動制御系のように，内部の要素の特性と要素間の結合を

示すブロック線図が与えられている場合，アナログ電子計算機を用いると，特性の解析や改善などをすみやかに行うことができる。その場合，演算回路が有効に利用される。

さらに，実際の装置の動作を調べたり，航空機の飛行特性や，原子炉の反応特性をシミュレーションしたりすることができる。このようにして用いられるアナログ電子計算機を**シミュレータ**という。シミュレータは，実際の装置を使っての実験や，操縦訓練よりはるかに経済的で安全である利点がある。

問　題

1. アナログ電子計算機の主な構成部分を挙げ，その働きを説明せよ。

2. 図4で，点AおよびCに解記録部を接続すれば，何が観測されるか。

3. アナログ電子計算機の種類を挙げ，その特徴を述べよ。

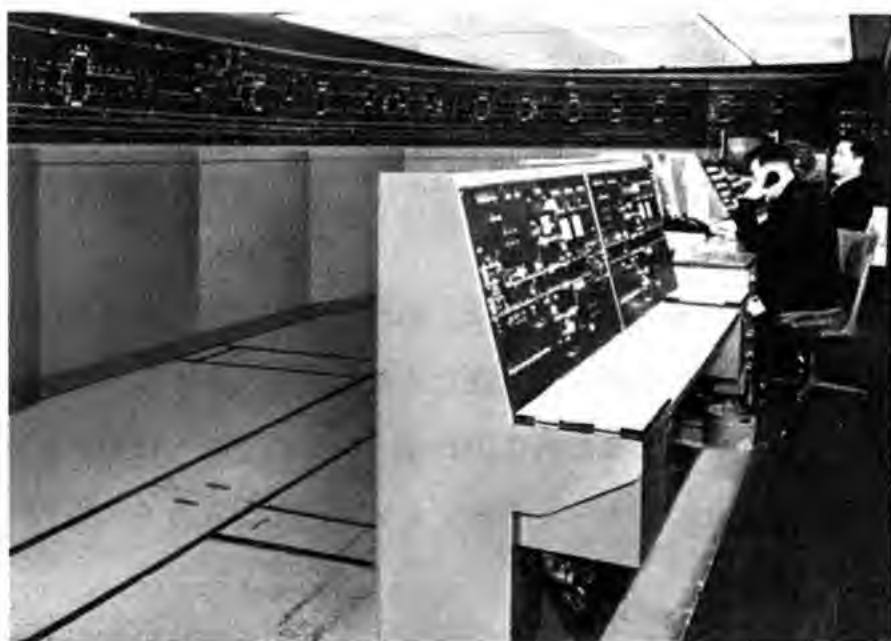
4. アナログ電子計算機の特徴を，ディジタル電子計算機と比較して述べよ。

5. アナログ電子計算機の主な演算回路とその働きを説明せよ。

第11章 自動制御

自動制御は，オートメーションの中心をなす工学であると考えてもよい。オートメーションの応用は，家庭から銀行・工場さらに，ロボット・宇宙船にいたるまで，非常に広い範囲にわたっており，近代社会は，オーメーションなしには，なりたたないといってもいいすぎではない。

この章では，自動制御とは何か，オートメーションの中で自動制御がどのような役割を果たしているかについて調べる。

鉄道管制システム

オートメーションと自動制御

この節のねらい　ここでは，まずオートメーションが社会において果たしている役割について述べ，次にこれを支えている自動制御の技術，すなわちシーケンス制御とフィードバック制御について学ぶことにする。

1. オートメーション

オートメーションは，装置・機器を一定の目的のために組み合わせて，自動化されたシステムを構成することであり，自動制御は，それを実現するさいに必要な技術の中心となるものである。

自動制御の技術は，第1次産業革命のころからあった。例えば，18世紀末，ジェームス=ワットによって発明された蒸気機関の調速機は，その一例である。それらがオートメーションという形の中で意識的に利用され始めたのは，自動車産業・石油工業で製品の大量生産をめざした，1930年代のアメリカ合衆国においてであった。自動制御として体系化が進んだのは，第2次世界大戦中から戦後にかけてである。

簡単なオートメーションは，われわれの身近なところに見られ，空調器・電気冷蔵庫・電気洗たく機などは，多かれ少なかれ自動化が行われている。また外に出れば，交通信号機・自動エレベータ・自動販売機など，オートメーションを応用した多くの実例を見ることができる。

しかし，現在，オートメーションが最も重要な役割を果たしてい

表1　オートメーションの効果

利用側	効果
製品	品質の向上 価格の低減
生産設備	生産速度・生産能率の向上 生産設備の信頼性・安全性の向上 省資源・省エネルギー
従業員	危険な作業からの解放 労働時間の短縮 省力化（労働力の節減）

るのは，工場の生産設備であって，近代工場といわれる工場では，いたるところでオートメーション化が見られ，大工場全体がほとんど無人に近い状態で運転されている例もまれではない。

このような工場のオートメーションは，われわれの身近にあるものに比べるとはるかに複雑であり，かつ精巧にできており，それの果たしている役割はきわめて大きい。その主なものを挙げると，表1のようなものがある。

オートメーションは非常に広い範囲にわたっているが，それを大別すると表2のようになる。

表2　オートメーションの種類と特徴

オートメーションの種類	用途および特性
プロセスオートメーション	薬品・燃料・肥料・砂糖など，形がないか，形自体が本質でない製品を作る場合で，フィードバック制御が広く活用されている。
メカニカルオートメーション	家庭電器・自動車・工作機械など，形自体が重要な役割をもっている製品を作る場合で，シーケンス制御が広く活用されている。
オフィスオートメーション	会計・計算・記帳・読み取り・書類作成など，いわゆる事務のオートメーションで，これには電子計算機が主役として使われている。

電子計算機は，現在では，オフィスオートメーションに限らず，プロセスオートメーション，メカニカルオートメーションにおいても広く用いられており，その結果，事務と生産を結び付けて一体とし，受注から生産，さらに出荷までを漸次自動化しようとする方向にある。

問 1. 現在，社会で，オートメーションが進行している理由を考えてみよ。

問 2. オートメーションが進んでいると思われる産業を具体的に挙げよ。

2. 制御と自動制御

(1) 制 御

ある目的に適合するように，対象となっているものに所要の操作を加えることを**制御**(control)という。

例えば，図1のように，対象である直流電動機を回転させるという目的のためにはスイッチSを閉じ，停止させるにはスイッチを開くという操作をすればよい。これは一種の制御である。

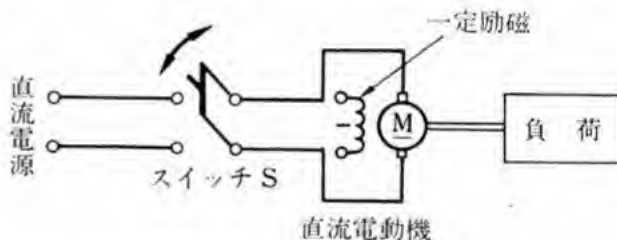



図 1 定性的な制御の例

また一方，回転している直流電動機の回転速度を，決められた値に一致させるという目的のためには，図2に示すように，その回転速度を回転計で測定し，その値が決められた値となるように，直流電動機の電機子端子電圧をすべり変圧器のしゅう動子によって増減させる操作を加えればよい。これは，上で学んだスイッチの開閉による制御とは異なるもう一つの制御である。

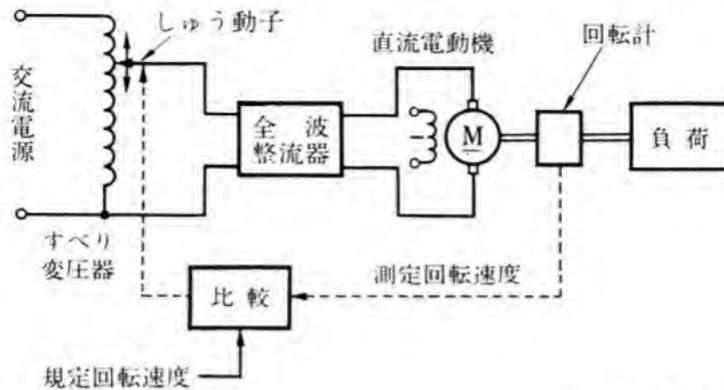



図 2　**定量的な制御の例**

これには，いずれも制御の対象となるものがあるが，これを**制御対象**（controlled system）といい，制御することが目的となっている制御対象内の量を**制御量**（controlled variable）という。

制御を行うには，目的に対応する命令が外から加えられているはずであるが，これを**制御命令**という。

図1および図2の制御の例のうち，前者では，制御量をいくつかある状態（例えば停止・駆動状態）のうち，どの状態にするかが制御の目的である。その意味でこのような制御を**定性的な制御**という。これに対して後者の例は，制御量の値（例えば回転速度の値）を決められた値にするような制御を行っているので，**定量的な制御**という。定量的制御の場合には，制御命令は具体的な値をもつので，これをとくに**目標値**とよんでいる。

（2） 自　動　制　御

制御を人手を借りずに装置で行うことを**自動制御**（automatic control），自動制御を行う装置および制御対象全体を含めた一つの系統を**自動制御系**（automatic control system）という。

自動制御系では，制御命令自体は外から与えられ，それに従って自動制御が行われるが，大きな装置・系統の制御では，外から与え

られる命令は，装置全体の始動・連続運転・停止のような概括的なものであることが多い。このような場合には，外から与えられる概括的な命令と，制御対象の状態とから判断して，適時，適切な個々の制御命令を作り，命令する必要がある。ここで，外部から与える概括的な命令のことを**作業命令**または**入力命令**（input command）という。作業命令と制御対象の状態から判断して，適時，適切な制御命令を作ることを**命令処理**，これを行う部分を**命令処理部**ということにする。

自動制御系では，単に制御を行うだけでなく，それに伴う命令処理も自動的に行わなければならないことが多い。

自動制御は，シーケンス制御とフィードバック制御に大別される。

（3）　シーケンス制御

図3は定性的自動制御の例を示したものである。この例では，スイッチS_cの開閉が制御命令であり，この信号のとおりに接点S_mが人手によらないで開閉されている。すなわち，自動制御の形を備えている。一方，この例において，S_cの開閉は作業命令であると考えることもでき，それがそのまま制御命令となっている。このように，命令処理を伴わない定性的自動制御はきわめて簡単である。

これに対して図4(a)は，図3の定性的自動制御系にタイマを付加したものである。この系の動作は，S_0を投入してからタイマにセットされた一定の時間だけS_cが閉じて電動機が回転し，その後は停

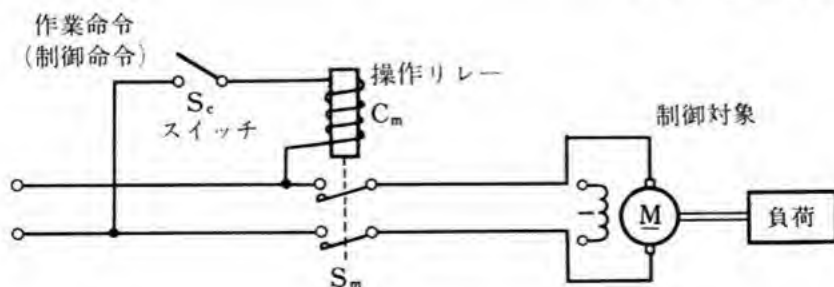
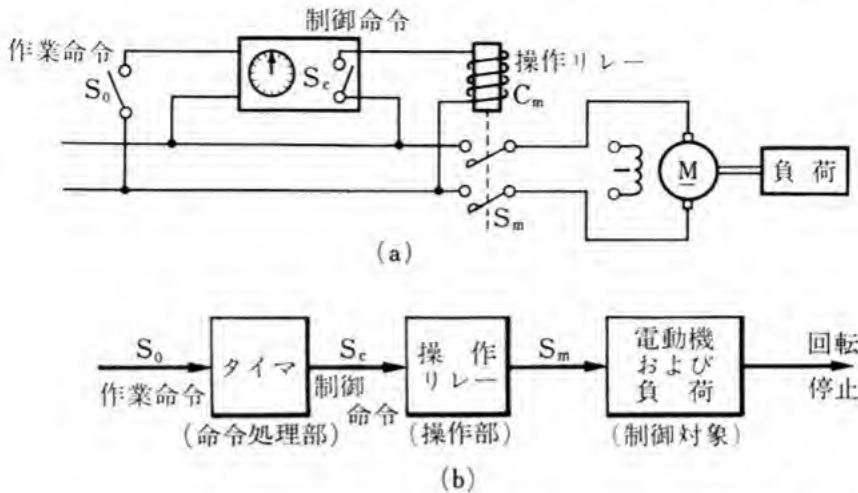



図3　定性的自動制御の例

図 4 命令処理を行っている定性的自動制御の例

止するようになっている。この自動制御の動作機構を原因・結果のつながりで示したのが図(b)である。この場合，S_cは開閉制御命令であり，作業命令とは異なっている。すなわち，この系ではタイマによって作業命令と制御命令との間の命令処理が行われている。一般に命令処理は，あらかじめ定められた順序・条件によって行われ，その結果として制御命令を発生する。

このように，あらかじめ定められた順序，または条件によって，制御の段階を逐次進めていく定性的な自動制御を**シーケンス制御**(sequential control)といい，これが行われている装置・機械全体を**シーケンス制御系**という。したがって，シーケンス制御系は，定性的制御を行う部分と，命令処理を行う部分とからなりたっている。

(4) フィードバック制御

図2の直流電動機の場合，制御命令すなわち回転速度の目標値を一定のプログラムに従って時間的に変化させたとする。これに従って電動機を制御するには，すべり変圧器のしゅう動子の位置を目標値の変化に応じて動かせばよいが，実際にはそれだけで制御の目的

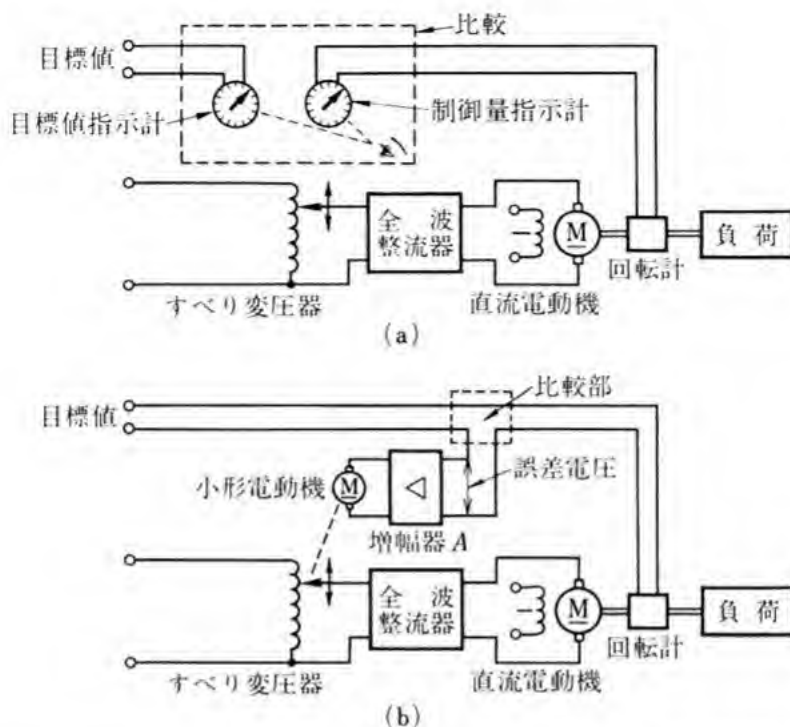



図 5 定量的制御の自動化の例

が完全に達成されるとは限らない。例えば，負荷や界磁電流が変動すると，かりに，しゅう動子をある位置に固定していても，電動機の回転速度は変動する。

したがって，電動機の回転速度を目標値どおりにするためには，まず図5(a)のように，制御量の指示計と，目標値の指示計をつけ，両者の指示を比べ，その結果，制御量が目標値より低い場合には回転速度を上げる向きに，高い場合には下げる向きにしゅう動子を動かし，これを常時続ける必要がある。

図(b)は，この作業を自動化する一つの方法である。すなわち，系の外から与えられる目標値と，回転速度の測定値とを比較部で比較し，その差を刻々求め，それを増幅して小形電動機に加えてしゅう動子を動かし，これによって電動機の回転速度を絶えず目標値に一致させるようにしている。図6は，その動作機構を図で示したも

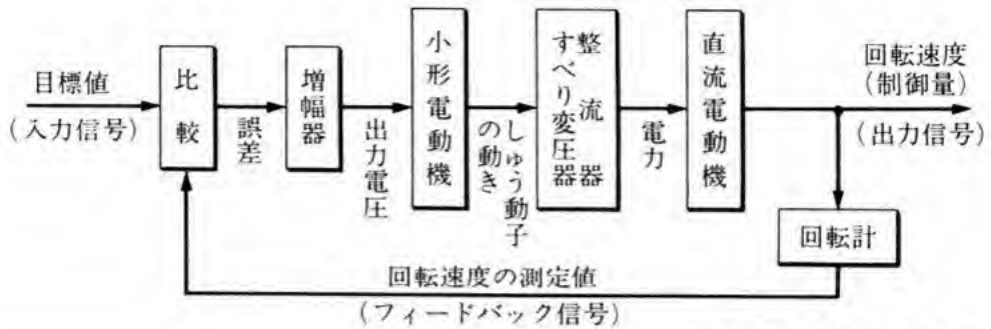
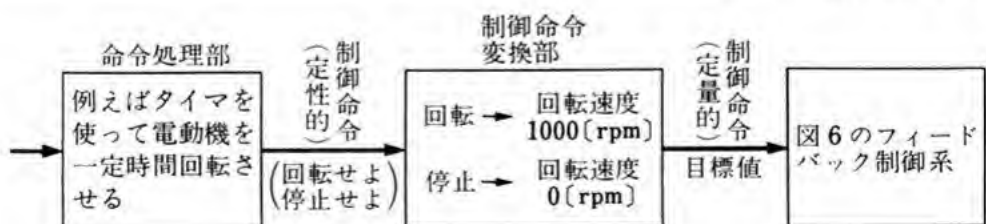



図 6 自動化の動作機構

のである。

図から明らかなように，この系では，制御量である回転速度から得た信号（出力信号）を，目標値（入力信号）のところにもどして比較し，その間の誤差をなくす向きに制御が行われている。このように，出力信号を入力信号にもどすことを**フィードバック**（feedback）または**帰還**といい，とくに，上で学んだように誤差をなくす向きのフィードバックを**負フィードバック**という。負フィードバックによって，目標値どおりに制御を自動的に行うことを，**フィードバック制御**（feedback control）という。

フィードバック制御では，フィードバックを行っているために系が閉じたループを形作っているので，**閉ループ自動制御**（closed loop control）ということもある。これに対して，図3，図4の自動制御系では，系は閉回路とはならない。このような自動制御を，**開ルー**

図 7 制御命令の変換

プ自動制御（open loop control）という。

フィードバック制御では，ふつう命令処理を行う部分を考えず，制御命令である目標値から考える。したがって，図7に示すように，制御内容あるいは制御対象などの条件によって，どのような種類の目標値を与えるかを判断する命令処理部をもつ場合，フィードバック制御系を内部に含むシーケンス制御系になることがある。この場合，定性的な制御命令を定量的な制御命令に変換する部分が必要となる。

問 3. われわれの周辺で，シーケンス制御とフィードバック制御が行われている代表的な装置・機器の例を挙げよ。

問　　題

1. オートメーションには，大別するとどのようなものがあるか。
2. 銀行で行われているオートメーションは，どのような種類に属するか。
3. 定性的な制御と定量的な制御との違いについて述べよ。
4. 制御命令と作業命令とはどのように異なるか。
5. フィードバック制御の特徴は何か。
6. 発電機の電圧を負荷にかかわらず自動的に一定に保つようにしたい。どのような種類の制御によればよいか。
7. 家庭電化製品の例について，それらに採用されている制御が，フィードバック制御かシーケンス制御かを調べてみよ。
8. 電気洗たく機の制御の手順を調べよ。また，洗たくを完全に自動化するには，どのような手順を自動化しなければならないか。

シーケンス制御

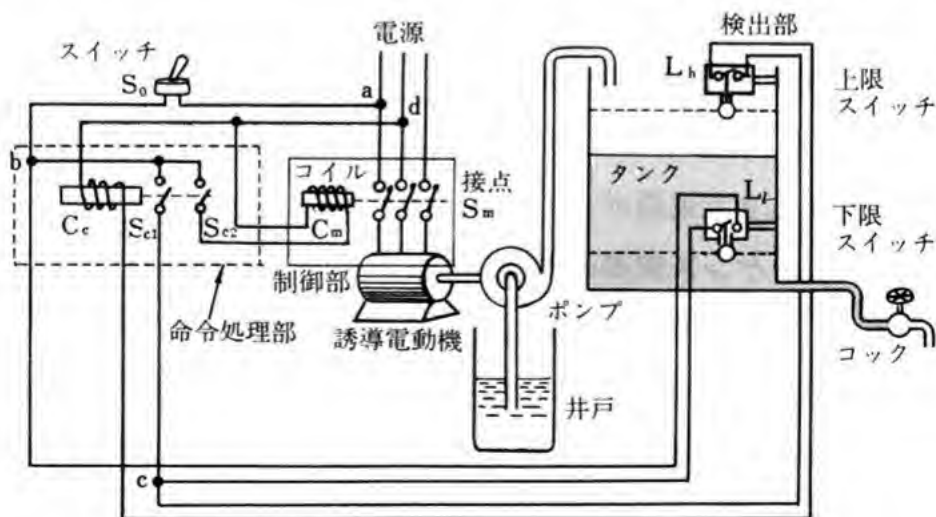
この節のねらい　簡単なシーケンス制御は，電気洗たく機・自動販売機・交通信号機・ネオンサインなど，われわれの身近なところに多く使われている。一方，エレベータ・コンベヤ・工作機械・産業用ロボットなどの装置や発電所・自動車工場などの大きなシステムの自動化にもシーケンス制御はなくてはならないものである。

実際のシーケンス制御系は，その目的や制御対象の種類・規模などによって，きわめて簡単なものから非常に複雑なものまで多種多様にわたっている。しかし，それを構成している要素や要素の基本的な組み合わせ方は，それほど複雑ではない。

1. シーケンス制御系の動作と構成

(1) シーケンス制御系の動作

図1は，シーケンス制御の一例を示したものである。この系の目

図 1　シーケンス制御の例

的は，井戸水をタンクにくみ上げ，コックからつねに水が出せるようにすることである。すなわち，上限スイッチ L_h および下限スイッチ L_l によって，タンクの水位に上限と下限とを定め，水位が下限に達したとき電動ポンプが始動し，上限に達すると電動ポンプが停止して，水位が下限になるまで休止しておくようにする。

いま，スイッチ S_0 を閉じて作業命令を与えておき，タンクの水の使用によって水位が下限まで低下すると，下限スイッチ L_l の接点が閉じる。このため，点 a から，スイッチ S_0，点 b，L_l の接点，点 c，上限スイッチ L_h の接点，コイル C_e を通って，点 d までの間が導通するから，コイル C_e が電源に接続されて励磁され，接点 S_{e1} および S_{e2} が閉じる。接点 S_{e2} が閉じるとコイル C_m が励磁され，接点 S_m が閉じて電動ポンプが始動し，タンクの水位が上昇する。タンクの水位が上昇すると，下限スイッチ L_l の接点は開くが，L_l と並列の接点 S_{e1} が閉じているので，電動ポンプは運転を続ける。

水位が上限に達すると，上限スイッチ L_h の接点が開いてコイル C_e が消磁されるので，接点 S_{e1} および S_{e2} が開く。したがって，コイル C_m が消磁され，接点 S_m が開き，電動ポンプは停止する。

水が使われて水位が下がると，上限スイッチ L_h の接点が再び閉じるが，水位が下限に達するまで L_l および S_{e1} は開いている。したがって，C_e は励磁されないから S_m は開いたままであり，水位が下限に達するまで電動ポンプは休止の状態を続ける。

（2） シーケンス制御系の構成

図２は，図１の系の動作機構を一般化して示した図である。図において，長方形の箱はシーケンス制御系の各構成部分に対応し，矢印のついた線は信号を示す。なお，信号を ⇨ で示してあるのは，信号の数が１個だけとは限らない場合を意味する。以下，図２の各

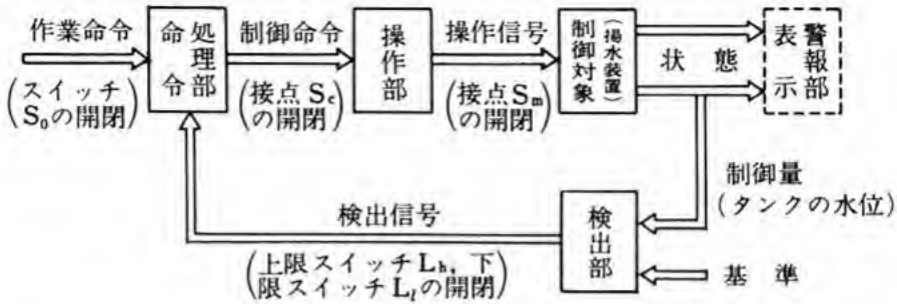



図 2　一般化されたシーケンス制御系の動作機構

部の構成について調べる。

制御対象　これは，制御しようとする目的の装置または機械のことで，図1の系では，電動ポンプ・井戸およびタンクからなるいわゆる揚水装置に相当する。なお，**制御量**は，制御対象において制御しようとする目的の量のことで，この例ではタンクの水位がこれに相当する。

検出部　これは，制御量が所定の状態であるかないかを示す2値信号を発生する部分であり，図1の系では，上限スイッチL_hと下限スイッチL_lとが検出部を構成する。なお，検出部で検出された信号を**検出信号**という。

命令処理部　これは，すでに学んだように，作業命令・検出信号や，あらかじめ設定してある条件・時限などから制御命令を作る部分であり，図2では，作業命令と検出信号とから制御命令を作っている。

操作部　これは，制御命令を増幅したり，変換したり，安全対策を施したりして制御対象を直接制御する部分である。図1の系で，接点S_{c2}は電流容量が小さいので，電動機の電源回路を直接開閉することはできない。そこで，コイルC_mによって電流容量の大きい接点S_mを開閉するようになっている。すなわち，接点S_{c2}から接点S_mまでが操作部であり，接点S_mの開閉が操作である。

図2に示した基本的な構成部分のうちの，あるものが欠けている場合もある。例えば，p. 106 の図3の系では検出部および命令処理部がなく，p. 107 の図4(a)の系では検出部が欠けている。

問 1. 図1のシーケンス制御系で，上限スイッチを用いず，タイマを利用することによって，ほぼ同じ制御を行わせるには，どのようにすればよいか。

問 2. 電熱器によって湯を沸かす作業を，シーケンス制御によって自動化することを考えよ。

2. シーケンス制御用機器

(1) スイッチ

命令スイッチ　命令スイッチは，手動操作スイッチともよばれ，作業命令を与えたり，命令処理の方法を変更したりするために使われる。命令スイッチは，その動作機能から，復帰形のものと保持形のものとに分けられる。

復帰形命令スイッチは，人が操作を加えるときだけ接点の開閉状態が変わり，操作をやめると元に復帰するものである。図3(a)は工業用の復帰形命令スイッチの例で，押しボタンスイッチである。図(b)

図 3　復帰形命令スイッチの例

(a) 押しボタンスイッチ

(b) 図記号

(a) 両押しボタンスイッチ

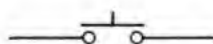
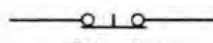
(b) セレクタスイッチ

(c) 図記号

図 4　保持形命令スイッチの例

は押しボタンスイッチの図記号で，図(b)①は操作するときだけ接点が閉じ，図(b)②は操作するときだけ接点が開くスイッチである。

一般に，操作しているときだけ閉じる接点を**a接点**（a contact）または**メーク接点**（make contact）といい，逆に，操作時だけ開く接点を**b接点**（b contact）または**ブレーク接点**（break contact）という。

保持形命令スイッチは，いちど操作すれば，反対の操作を行うまで，その接点の開閉状態をそのまま保つものである。図4は保持形命令スイッチの一例で，図(a)は両押しボタンスイッチ，図(b)はセレクタスイッチである。図(c)は，保持形命令スイッチの図記号を示したものである。

検出スイッチ　検出スイッチは，制御対象の状態または変化を検出するためのスイッチで，実際には，位置・液面・速度・温度・圧力・電圧などの量を検出するために使われる。

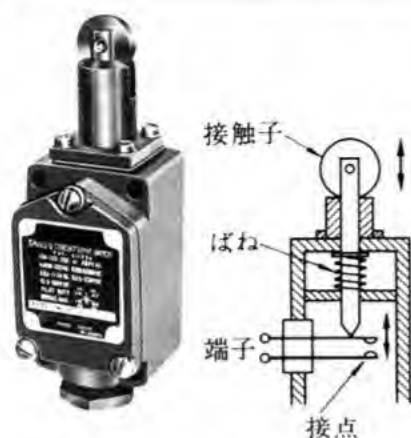
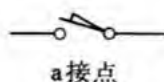
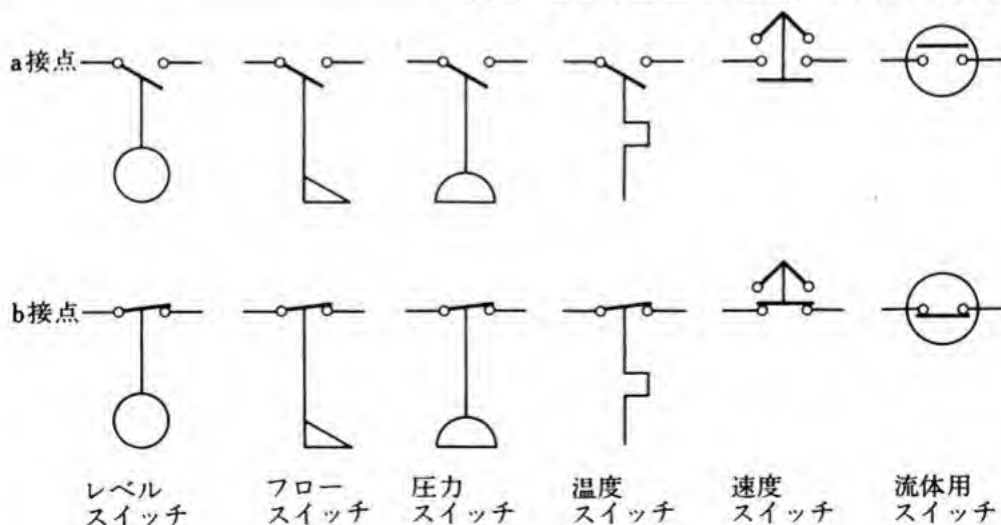
図5(a)は，位置の検出に広く用いられる**リミットスイッチ**の例である。その接触子に物体が接して接触子が働き，接点が開閉する。図(b)は，リミットスイッチを表すときに使われる図記号であり，図(c)は，リミットスイッチ，その他，機械的な操作で開閉される

(a) リミットスイッチ　(b) リミットスイッチの図記号　(c) 検出スイッチの一般的図記号

図 5 検出スイッチの例

検出スイッチの一般的図記号である。位置以外の物理量の検出でも，その物理量をまず変位または力に変換し，リミットスイッチを動作させるものが多い。図6は，浮きとリミットスイッチを組み合わせて液位を検出するレベルスイッチ

図 6 レベルスイッチ

図 7 各種の物理量の検出スイッチの図記号

の例を示したものである。また，図7に，各種の物理量の検出スイッチの図記号（MAS：日本工作機械工業会規格，JIS）を示す。

（2） 継　電　器

コイルに入力信号として電流を流し，電磁力を発生させ，これによって，出力信号として接点を開閉する機能をもった装置を**継電器**（relay）といい，目的に応じて大小いろいろのものがある。

図8(a)は，主として命令処理用に使われる**制御用継電器**で，小形で接点数が比較的多く，4接点ないし32接点がふつうである。図(b)は，**電力継電器**で，命令処理用にも使われるが，600〔V〕，10〔A〕程度までの負荷の操作用として使われる。

図(c)は，継電器を一般的に示す図記号である。

図に示すように，コイルとそれによって働く接点の間の関係を表すのに，同一の文字記号または数字記号を用いるのが便利である。また，一つのコイルによって動作する接点がいくつかある場合，図のA_1, A_2のように，適宜添字ないし補助記号をつけ，さらにb接点には文字の上に“－”をつけて，$\overline{A}_3$, $\overline{A}_4$のように，それぞれ区別して表すのがふつうである。

図8 継　電　器

(a) 制御用継電器　(b) 電力継電器　(c) 図記号

(a) 限時動作式　　(b) 限時復帰式

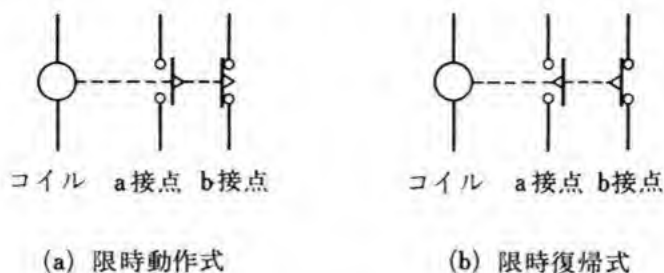



図9 限時継電器の図記号

(3) 限時継電器

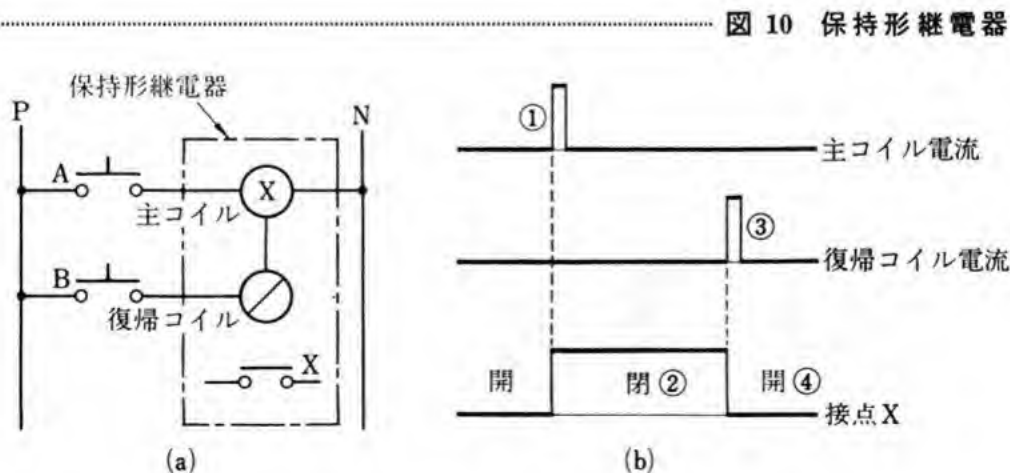
タイマまたはタイムリレーとよばれるもので，入力信号の変化時より，所定の時間だけ遅れて出力信号が変化するものである。

コイルに入力信号として電流を流した後，出力接点が一定時間後に動作するものを**限時動作式**，コイルの電流を切った後，出力接点が一定時間後に復帰するものを**限時復帰式**という。図9は，それらの図記号である。

(4) 保持形継電器

保持形命令スイッチと同じように，図10(b)の①，③のような入力信号に対して，接点がそれぞれ②，④のように開閉する継電器を**保持形継電器**（latch relay）という。

図10 保持形継電器

(a)　　(b)

すなわち，保持形継電器は，図(a)のように主コイルと復帰コイルをもち，接点Aを閉じ，図(b)①のように，主コイルに電流を流すと接点が②のように閉じ，それと同時に保持機構が働いて，主コイルの電流を切っても，接点状態はそのまま保たれる。この状態で接点Bを閉じ，③のように復帰コイルに電流を流すと，保持機構が外れて④のように接点が開く。開いた後は，復帰コイルの電流を切っても，接点状態は主コイルが励磁されるまでそのまま保たれる。

このような保持形継電器は，コイルに通電する期間が短いので，消費される電力が少なく，したがって，発熱も少ないので，小形のコイルですみ，大電流用の電磁接触器やしゃ断器にも応用される。

なお，図(a)で，P, Nは一般に電源端子を示し，Pは正の端子，Nは負の端子を表し，電源は直流であることを意味する。交流の場合にはP, Nの代わりに，U, Vを用いる。

(5) 過負荷継電器

過負荷継電器は，電流がある一定値を超えて一定時間以上回路を流れる場合，これを過負荷として回路を切る継電器であり，負荷電流が流れて熱を発生する抵抗発熱体とバイメタルとを組み合わせたもので，**熱動継電器**または**サーマルリレー**ともよばれる。過負荷継電器が働き，バイメタルが冷えた後で，セットボタンを押すと元の状態にもどる手動復帰形がふつうであり，その図記号を図11に示す。

図11 過負荷継電器の図記号

(6) ステッピングリレー

ステッピングリレーは，一定時間継続するパルス電流によって，複数の接点を順次切り換える一種の継電器である。

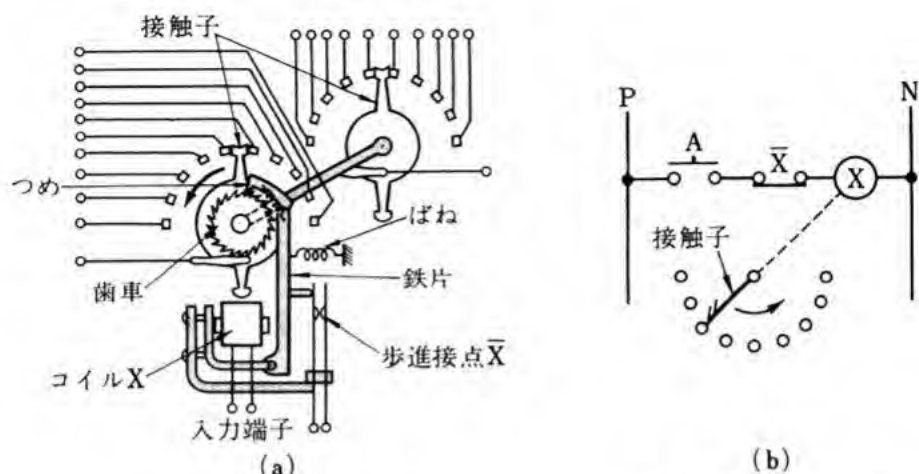



図 12 ステッピングリレー

図12(a)において，入力端子に電流が流れると，コイルXが励磁されて鉄片が吸引され，鉄片の先のつめによって歯車が1こまだけ進み，接触子が次の接点に移る。電流が切れると，ばねによって鉄片が元にもどされる。したがって，入力端子にパルス電流が流れると，そのパルスの数だけ接触子が歩進し，その数に相当する接点を選ぶことができる。

また，歩進接点$\overline{X}$を図(b)のようにコイルXに直列に接続すると，Aを閉じた場合，接触子は1ステップ進み，同時に$\overline{X}$が開いて鉄片は元にもどる。そうすると$\overline{X}$が閉じてもう1ステップ進む。すなわち，接触子はAが閉じている間，回り続ける。このような方法は，接触子を所定の位置にもどす場合に用いられる。

問 3. 図12(a)において，図(b)のような使い方をするには，接点Aをどこにつければよいか考えてみよ。

3. シーケンス制御のための基本回路

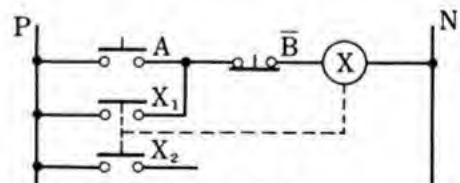
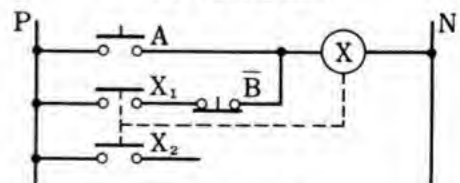
(1) 自己保持回路

保持形継電器と同じ動作をふつうの電磁継電器回路によって行わせることができる。このような回路を**自己保持回路** (selfhold circuit) という。図13(a), (b)は，それぞれ自己保持回路を示したものである。図において，Aは始動用押しボタンスイッチ，$\overline{B}$は停止用押しボタンスイッチである。図(a)において，Aが閉じると，コイルXが励磁されてX_1, X_2が閉じるので，Aが開かれた後もコイルXは励磁のまま保たれる。これは，図(b)の回路でも同様である。一方，$\overline{B}$が開かれると，図(a), (b)ともに，コイルが消磁されて元の状態にもどる。

しかし，Aと$\overline{B}$の押しボタンを同時に押すと，図(a)の場合には，コイルXは励磁されない。すなわち，復帰が優先される。これに対して，図(b)の場合，コイルXは励磁される。すなわち，動作が優先される。ふつうは，安全のために復帰優先の自己保持回路が使われる。

なお，自己保持回路では，停電によって，接点状態は元の状態に復帰するが，保持形継電器では，接点状態はそのまま保たれる点が異なる。

図14は，自己保持回路によるポンプ用電動機の始動・停止を行う回路例である。押しボタンスイッチPB_1を押す

図 13 自己保持回路

(a) 復帰優先形

(b) 動作優先形

と，自己保持回路が保持され，コイルMCの励磁で誘導電動機の電源回路が閉じ，始動される。その後，PB_1から手を離し，その接点が開いても電動機は回転し続ける。また，押しボタンスイッチPB_2を押すと，自己保持回路が解けて電動機は停止する。

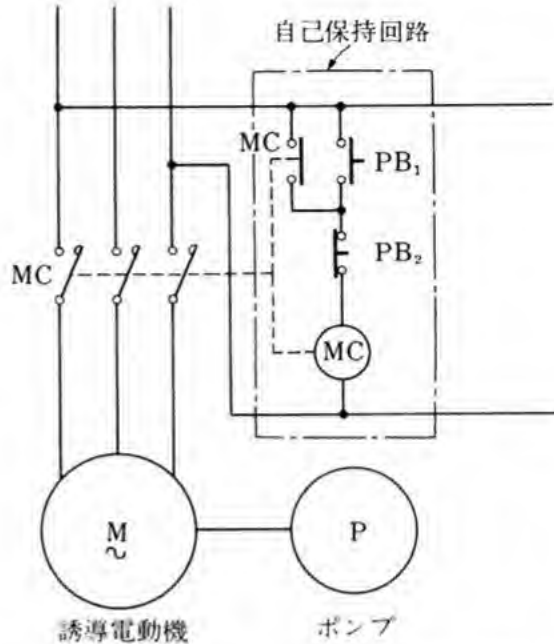



図 14　自己保持回路の例

（2）　始動渋滞（じゅうたい）検出回路

タイムリレーの代表的利用例として，装置に動作開始の入力信号が加えられた後，一定時間を経過しても正常な動作状態に入らない場合，その装置を停止させる回路を図15に示す。図15で，接点Aは動作開始と同時に閉じられる。これにより，限時動作式タイムリレーのコイルXが励磁されるが，タイムリレーで決められた一定時間前に正常な動作状態になったとき，これを検出スイッチ$\overline{B}$で検出して接点を開き，コイルXは消磁されて，タイムリレーは元の状態にもどる。しかし，所定の時間を経過しても接点$\overline{B}$が開かなければ，装置は異常であると判定して，タイムリレーの接点の動作によって，装置を停止させる。

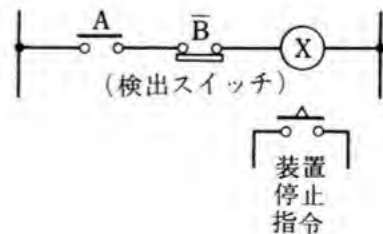



図 15　始動渋滞検出回路

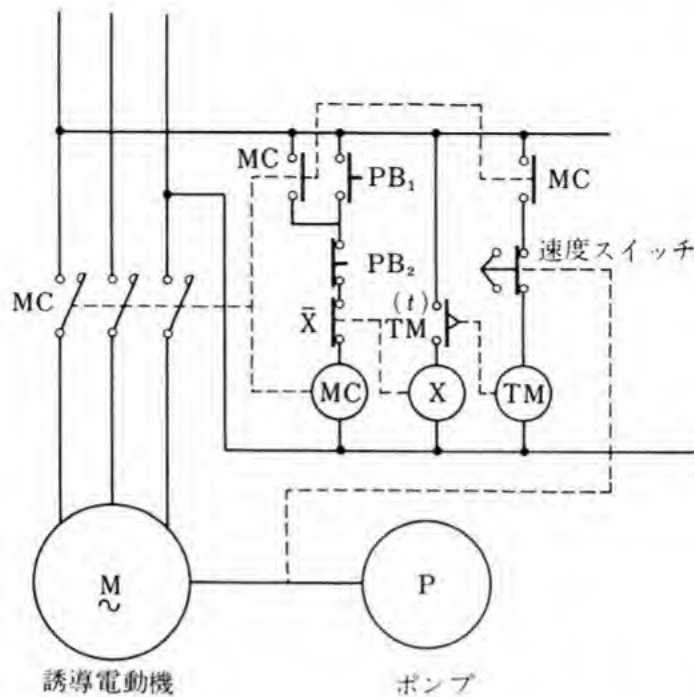



図 16　始動渋滞検出回路

図16は，この回路を用いた例である。自己保持回路により，押しボタンスイッチ PB_1 で誘導電動機を始動した後，タイムリレーの出力接点 TM が閉じる前に規定速度に達し，速度スイッチが開くとコイル X は励磁されることがないので，誘導電動機は回転し続ける。

しかしながら，ポンプの故障などにより，タイムリレーで決められた一定時間 t 以内で規定速度に達しない場合，接点 TM が閉じ，コイル X の励磁により接点 $\overline{X}$ が開き，したがって自己保持回路が復帰して誘導電動機は停止する。

このようにして故障を検知し，それがシステムにより大きな損害を与えないようにすることができる。

(3) 論 理 回 路

命令処理部では，検出信号や作業命令などに対して論理的な判断を行い，その結果に応じた制御信号を発生することがある。このように，入力信号について論理判断を行い，その結果を出力信号として得る回路を**論理回路**という。論理回路の基礎については「電気基礎」ですでに学んだが，論理和・論理積および否定を必要に応じて組み合わせて，上で述べた論理判断を行わせることから**組合せ回路**ともよぶ。図17は，組合せ回路の機能を明確にするため，ブロック線図で示したもので，この場合の信号はもちろん2値信号に限ら

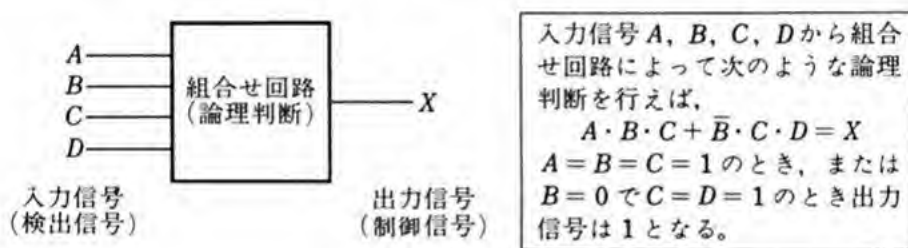
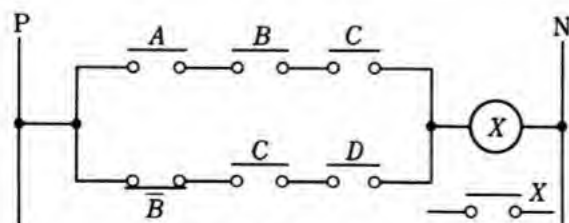
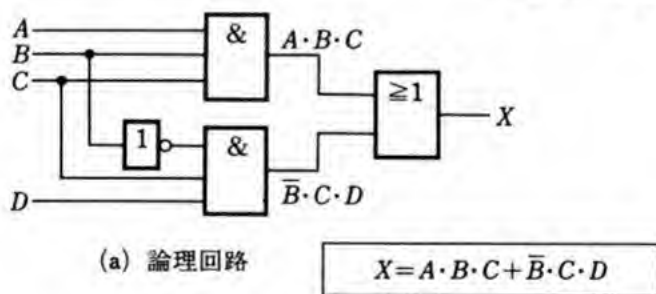
図 17 組合せ回路

入力信号 A, B, C, D から組合せ回路によって次のような論理判断を行えば，

$$A\cdot B\cdot C+\overline{B}\cdot C\cdot D=X$$

$A=B=C=1$ のとき，または $B=0$ で $C=D=1$ のとき出力信号は1となる。

図 18 組合せ回路の例

(a) 論理回路

(b) 接点回路

れる。例えば，図で出力信号 X が入力信号 A, B, C, D について次の式（1）の関係を満足する場合，X は A, B, C が同時に1のときか，あるいは B が0で，C, D が1のときに限り1となる。

$$X = A \cdot B \cdot C + \overline{B} \cdot C \cdot D \qquad (1)$$

5　また，図18(a)は論理素子を用いて，また図(b)は接点を用いて，上式を組合せ回路として構成したものである。

問 4. 図1に自己保持回路が含まれている。これは動作優先形か復帰優先形か。

問 5. 自己保持回路の継電器として，限時復帰式のタイムリレーを用い
10　た。どのような性質をもつ回路となるか。

4. プログラム制御

（1） プログラム制御とその種類

あらかじめ定められたプログラムに従って，制御の各段階を順次進めるシーケンス制御を**プログラム制御**という。

プログラム制御では，図19に示すように，命令処理部のプログラム設定部に，作業の内容・順序が前もってプログラムされ，これを歩進部から発生する歩進信号によって順次進めていくが，その進み方が，図(a)のように，時間的に決められたもの（例えば交通信号機）を**時限プログラム制御**，図(b)のように，前の制御の終了を信号として受け取って，これによって，次に行うべき制御の段階に進むもの（例えば自動販売機）を**順序プログラム制御**という。

（2） 時限プログラム制御

時限プログラム制御の歩進部は，タイマによって構成される。図20はその一例で，**モータタイマ**とよばれる。電動機Mによってセレクタを回転させ，タイミングディスク上に配置したプラグの接点に，順次接触させることにより，セレクタの回転速度とプラグの位置とによって決まる時点で，歩進信号を発生させる方法である。

モータタイマを用いた時限プログラム制御の一例として，図21の

図 19 プログラム制御の命令処理部

作業命令
設定部
プログラム
制御命令
歩進信号
歩進部
（タイマ）

(a) 時限プログラム制御

作業命令
設定部
プログラム
制御命令
歩進信号
歩進部
検出信号
（制御終了信号）

(b) 順序プログラム制御

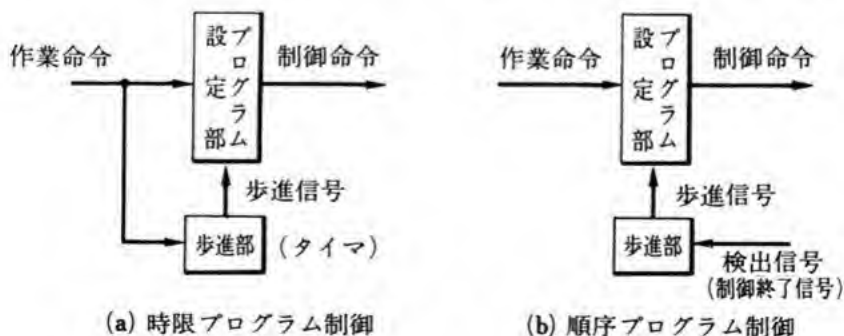
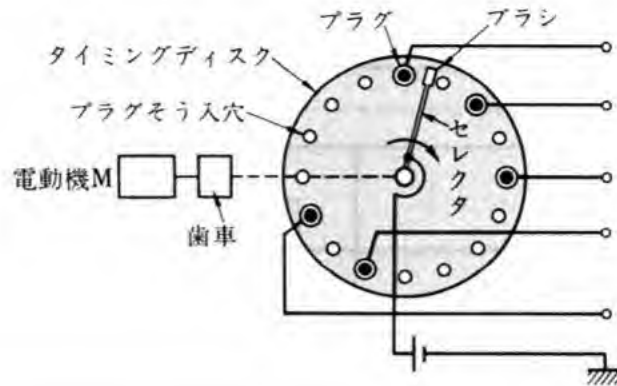



図 20　モータタイマ

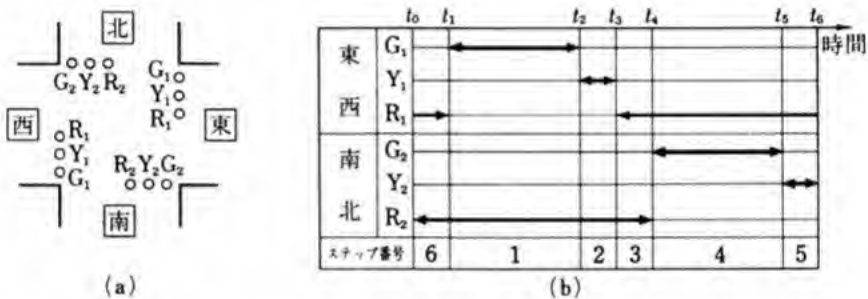



図 21　交通信号の制御プログラム

交通信号の例を考えてみる。図21(a)に示すように，東西南北の交差点に赤（R）・黄（Y）・緑（G）の信号灯が4組配置されているが，東西と南北は対(つい)となるので，東西の赤・黄・緑の信号灯をそれぞれ R_1, Y_1, G_1, 南北のそれぞれを R_2, Y_2, G_2 とする。これらは，図(b)に示すプログラムに従って点滅するものとする。

図で横軸は時間であり，太横線の部分が点灯期間を意味する。信号が切り換わる時点は t_1 から t_6 までであり，t_0 は1周期前の t_6 に相当する。これらの切り換え時点間を**ステップ**といい，t_1, t_2 間（東西の信号が緑）のステップを第1ステップとして，第6ステップまである。切り換え時点は，モータタイマのタイミングディスク上の，6個のプラグの相対位置で設定される。そこで，各切り換え時点 t_1,

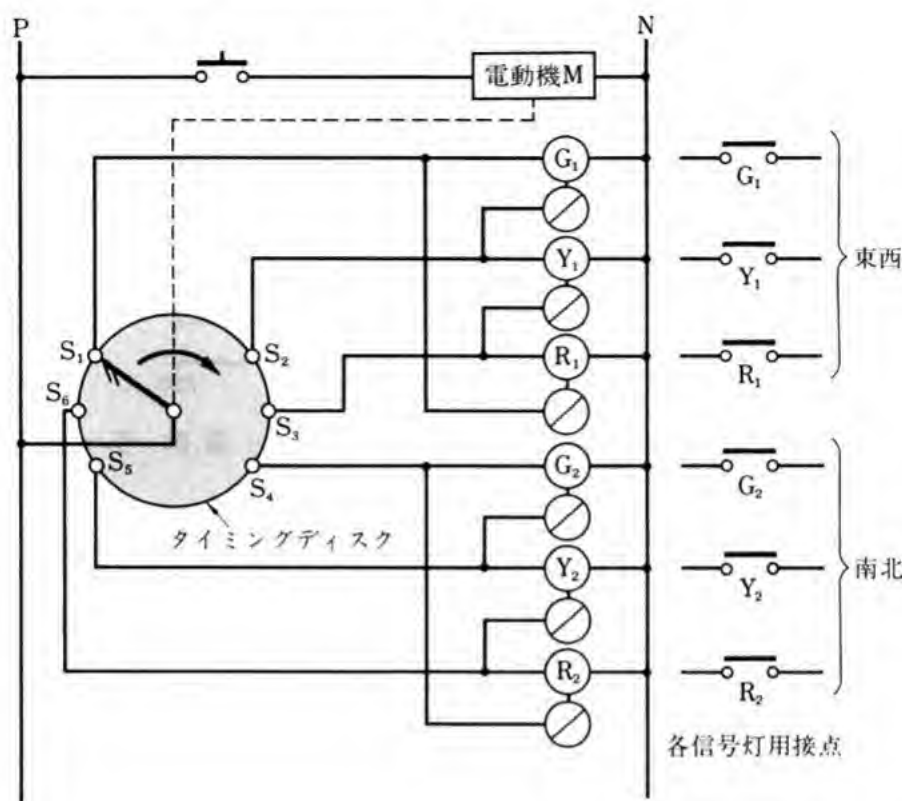



図 22 交通信号制御回路

t_2, ……, t_6 に相当するプラグの接点をそれぞれ S_1, S_2, ……, S_6 とすると，交通信号制御回路は図22のようになる。図では，保持形継電器が用いられ，それらの接点によって信号灯を点滅させるようになっている。図において，セレクタが S_6 の位置にきて R_2 が点灯した後，S_1 の位置に移動すると，それまで点灯していた R_1 が消灯して G_1 が点灯する。次に，S_2 の位置にくると G_1 が消灯して Y_1 が点灯する。同じようにして，S_3 の位置で R_1 が点灯した後，セレクタが S_4 に移動すると，それまで点灯していた R_2 が消灯して G_2 が点灯し，以後，S_5, S_6 と移動するに従って，Y_2, R_2 が順次点灯する。このようにして，図21に示したプログラムどおりに，信号灯の制御が行われる。

（3） 順序プログラム制御

順序プログラム制御では，プログラムの歩進は，各制御段階の終了の信号によって行われることが多い。図23は，位置①にある物体をリフトとコンベヤによって位置⑤まで順次搬送するシーケンス制御系で，順序プログラム制御の一例である。このための命令処理部には，いろいろな方式があるが，いま，図23に示すものを考えてみる。

初めに図23の①にあった物体を，なんらかの方法で②まで押し出したとすると，リミットスイッチL_1が閉じる。これによって，図24のように，接点1を閉じていた接触子①は，L_1が閉じることにより1ステップ歩進して接点2を閉じる。これと同時に，接触子②も接点1′から2′に進み，コイルDを励磁する。コイルDは，図23の接点Dを直接あるいは間接に閉じる（ふつうは，中間にもう1段，継電器を介し，一種の増幅を行う。ほかのコイルも同様である）。

その結果，電動機1はリフトを下降させるように回転する，物体が③の位置に到達し，リフト台の下面がリミットスイッチL_2を閉じると，ステッピングリレーは接点2, 2′から接点3, 3′に歩進する。

図 23 順序プログラム制御の例

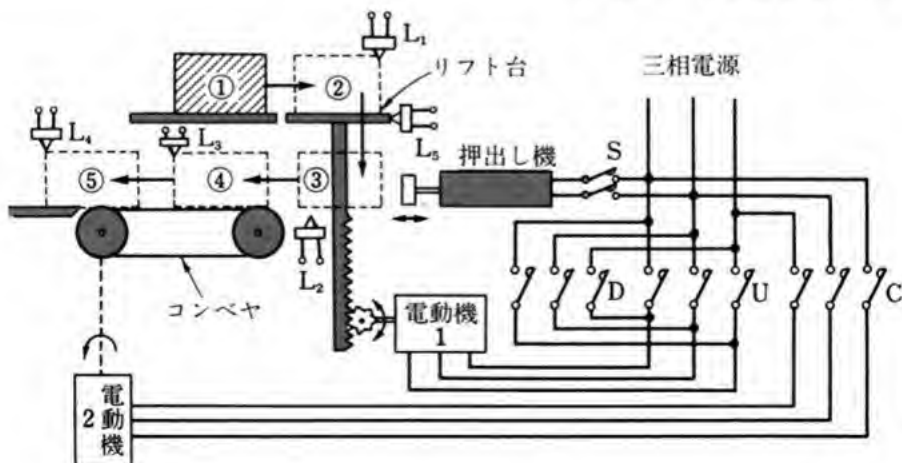
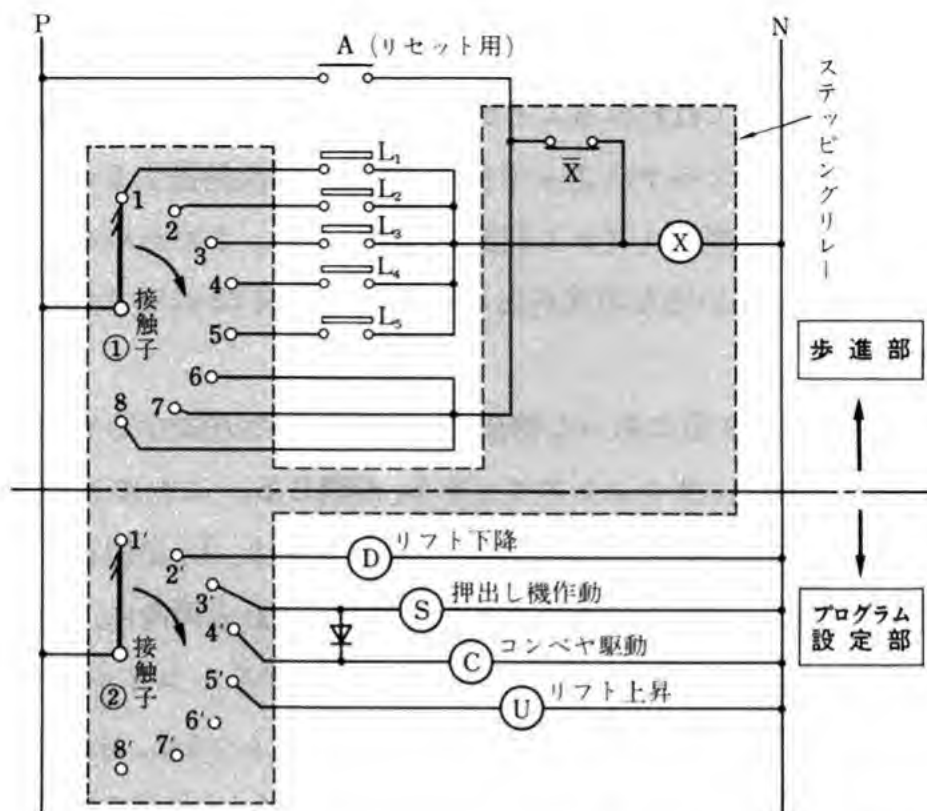



図 24　図23の命令処理部の例

その結果，コイルDは消磁されて図23の接点Dが開くので，電動機1が止まってリフトの下降は終わり，一方，押出し機作動用のコイルSおよびコンベヤ駆動用のコイルCが励磁され，図23の接点S, Cが閉じる。このため，押出し機により物体はコンベヤ上に送り出され，駆動されたコンベヤで物体は④の位置まで移動する。

④において，物体がリミットスイッチL_3を閉じると，ステッピングリレーは点3, 3′から4, 4′に歩進する。接点4′が閉じると，コイルCは前に続いて励磁されているが，コイルSは，コイルC，コイルS間に接続されたダイオードにより電流が阻止され，消磁されて接点Sが開き，押出し機への電圧は0となる。押出し機は，電圧を加えないとき，ばねにより元に復帰するものであるとすると，押

出し機の腕は元にもどり，物体はコンベヤにより ⑤ の位置まで移動される。

ここで，物体がリミットスイッチ L_4 を閉じると，ステッピングリレーは接点 4, 4′ から 5, 5′ に歩進する。このため，コンベヤ駆動用コイル C は消磁されるのでコンベヤは停止し，同時にリフト上昇用コイル U が励磁され，リフトは図 23 のリミットスイッチ L_5 を閉じる位置まで上昇する。リミットスイッチ L_5 を閉じると，ステッピングリレーは接点 6, 6′ に歩進し，コイル U が消磁されリフトは元の位置に停止する。一方，ステッピングリレーは接点 6 から 7, 8 と自己歩進し 1 へもどる。このようにして一連のシーケンス制御が終わり，新しい物体の到着まで待機状態を保つ。

問 6. プログラム制御の種類とそれらの特徴について述べよ。

問 7. ネオンサインの点滅の制御は，どのようなシーケンス制御に属するか。

5. 条件制御

（1） 条件制御の命令処理部

条件制御の命令処理部では，検出信号の組み合わせについて，前もって決められた条件に従って論理的な判断を行い，その結果に応じた制御信号を発生するもので，プログラム制御における命令処理部の働きと異なっており，一般に複雑な論理回路や，フリップフロップのような記憶要素から構成されている。

（2） 条件制御の例

図 1 では，揚水装置の最も基本となる部分だけについて考えたが，実際には，次のことも考える必要がある。

1)　井戸の水位が吸水管の下端より下がると，ポンプは空転を続けるので，一定水位以下では，電動ポンプの運転を中止する必要がある。

2)　電動機の焼損を予防するためには，電動機の電流が一定時間以上許容値を超えた場合に，電源を切る必要がある。

3)　不測の場合を考慮して，自動制御を手動制御に切り換え，手動スイッチによって，始動・停止ができる必要がある。

このような必要性から，上記の状態を検出する検出スイッチおよび手動自動切換スイッチが必要である。図25にそれらの配置を示す。

図において，Sは手動自動切換スイッチ，$\overline{L_s}$は井戸の水位を検出するレベルスイッチの接点であり，井戸の水位が規定値以下になったときに接点が開く(b接点)ようになっている。$\overline{O_h}$は，電動機に過大な電流が一定時間以上流れたとき開く過電流継電器の接点である。

この場合の命令処理部，すなわち，各入力信号に対する論理回路は，図26(a)のようになる。図でフリップフロップは，L_hが0であってもL_lがパルス状に1となれば出力信号L_tは1となり，それ

図 25　揚水装置を制御するための検出器・手動スイッチの配置

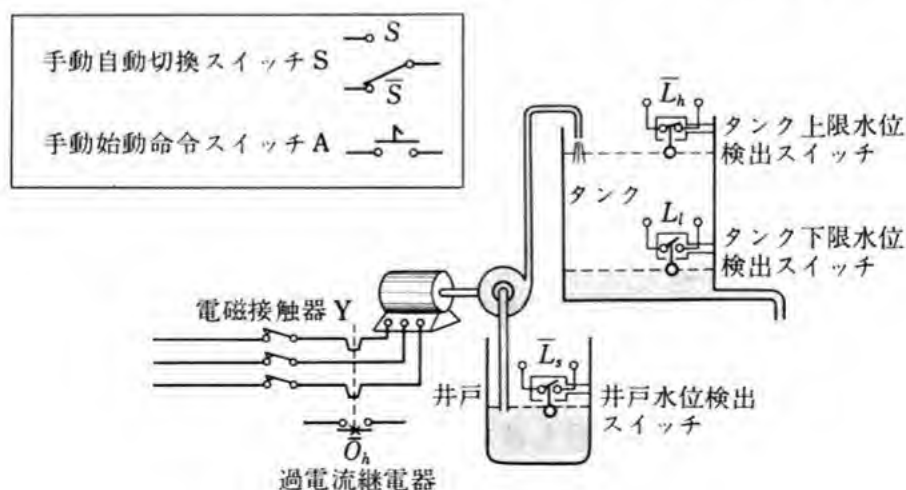
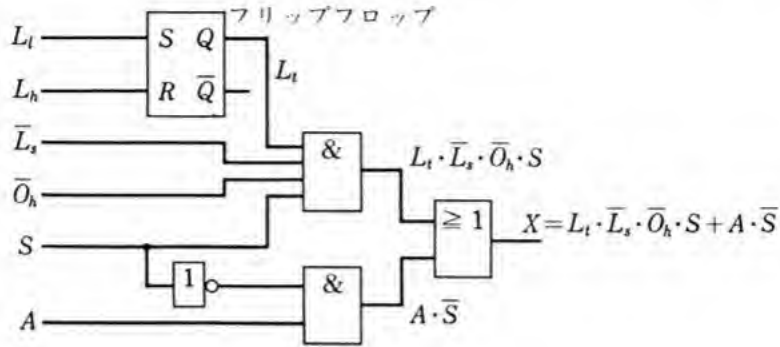
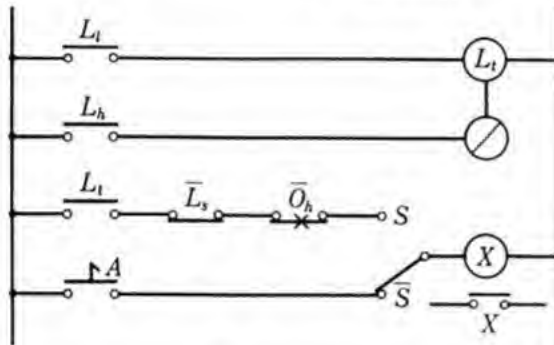



図 26　図25の命令処理部の組合せ回路

を記憶する。続いて L_l が 0 となって L_h がパルス状に 1 となれば L_t は 0 となる。いま，図 26 (a) で，X を図 25 の電磁接触器 Y の接点を閉じるための出力信号とすると，X が 1 となるためには $\overline{L}_s=1$（正常水位），$\overline{O}_h=1$（正常負荷），$S=1$（自動運転）および $L_t=1$（L_l の接点がいちど閉じ，現在揚水中の状態）の各条件が同時に満足するか，または $S=0$（手動運転）および $A=1$（手動始動命令スイッチが閉じた状態）の条件が満足すればよいわけである。これは論理式で表すと，次のようになる。

$$X=L_t \cdot \overline{L}_s \cdot \overline{O}_h \cdot S + A \cdot \overline{S} \qquad (2)$$

また，図 26 (a) の論理回路を接点回路で構成すると，図 (b) のようになり，図の保持形継電器は，フリップフロップに相当する。

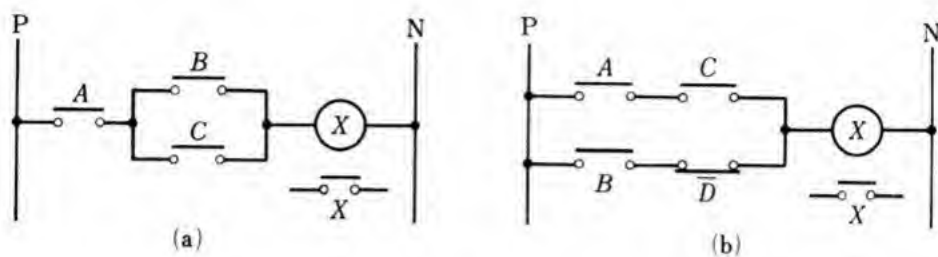



図 27

問 8.　図 27 の接点回路を論理回路で表せ。

問 9.　図 26 (b) で，保持形継電器を用いることができないとすると，ふつうの継電器を用いてどのような回路にすればよいか。　5

問　題

1．けい光灯スタンド用スイッチと白熱電灯用スイッチとは，機能的にどのような違いがあるか。

2．図28の回路を論理回路で示せ。

3．復帰形押しボタンスイッチを用いて，電熱器の点滅を制御する回路を考えよ。

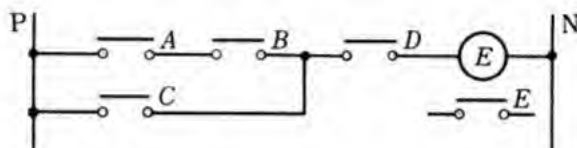



図 28

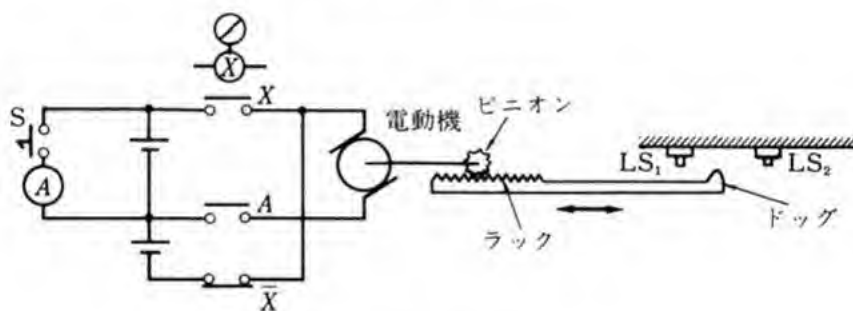



図 29

4．図29において，保持形押しボタンスイッチSを押すと，電動機・ラック・ピニオンによって直線運動をする部分の先端の突起部（ドッグ）が，二つのリミットスイッチLS_1, LS_2の間を繰り返し往復するようなシーケンス制御回路を考案せよ。ただし，保持形継電器を用いてもよい。また，$X=1$のとき，ドッグは右向きに動くものとする。

5．図24の順序プログラム制御をステッピングリレーによらずに実現してみよ。ただし，保持形継電器は利用できるものとする。

3 フィードバック制御の基礎

この節のねらい　先に，フィードバック制御とはどのようなものか，そのあらましについて学んだ。ここでは，フィードバック制御系の基本的な構成や動作，またその特性の表し方はどのようなものか。さらに，特性を良くするにはどのような方法がとられるかについて調べることにする。

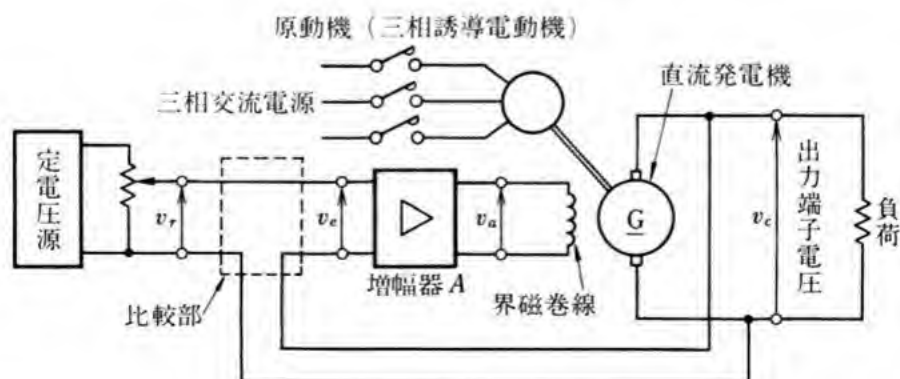
1．フィードバック制御系の動作と構成

(1) フィードバック制御系の動作

第1節で学んだように，定量的な制御を行う場合，制御した結果が正確に目標値（**制御命令**）のとおりにならないことが多い。このような場合，フィードバック制御が必要となる。

図1は，フィードバック制御系の例である。

この系の目的は，直流発電機の出力端子電圧を，原動機の速度変動や負荷の変動などにかかわらず，いつでも目標値としての電圧 v_r に一致させようとすることである。この例によって，フィードバッ

図1　直流発電機の出力端子電圧のフィードバック制御

ク制御の原理を簡単に数式で取り扱うことにする。

電圧 v_r は，定電圧源の電圧をポテンショメータで分圧して得られる電圧である。v_c は，出力端子電圧で制御量である。増幅器に加わる電圧 v_e は，次の式のように，目標値 v_r から制御量 v_c を差し引いたものである。

$$v_e = v_r - v_c \qquad (1)$$

また，増幅器の増幅度を A とすれば，その出力電圧 v_a は次のようになるものとする。

$$v_a = Av_e \qquad (2)$$

また，直流発電機に加わる原動機速度や負荷に変動などがない場合には，v_c は界磁巻線に加わる電圧 v_a に比例し，次の式がなりたつものとする。ただし，G は比例定数である。

$$v_c = Gv_a \qquad (3)$$

これらの式から，v_a, v_e を消去して v_r と v_c の関係を求めると，次のようになる。

$$v_c = \frac{AG}{1+AG}v_r = \frac{1}{1+\frac{1}{AG}}v_r \qquad (4)$$

式（4）から，AG の値が1に比べてきわめて大きいとすれば，出力端子電圧 v_c は，目標値 v_r にほぼ等しくなることがわかる。したがって，目標値 v_r を変化させれば，これに追従して制御量としての v_c も同じように変化する。

次に，原動機速度や負荷に変動がある場合について考える。いま，発電機をフィードバック制御を行わないで運転しているときの，原動機速度あるいは負荷の変動による出力端子電圧の変動を v_d とすれば，フィードバックがある場合には，式（3）の代わりに，次の式がなりたつ。

$$v_c = Gv_a + v_d \qquad (5)$$

したがって，式（5）と式（1），（2）から，v_c は次のようになる。

$$v_c = \frac{AG}{1+AG}v_r + \frac{1}{1+AG}v_d \qquad (6)$$

すなわち，式（6）の第2項は，負荷の変動などによって出力端子電圧が受ける影響を示す値であって，フィードバック制御を行わない場合に比べて，$\frac{1}{1+AG}$ に減少していることがわかる。

以上のことから，フィードバック制御には，次の二つの機能があることがわかる。すなわち，その一つは，図1の例で学んだように，目標値を変化させたとき，制御量である出力端子電圧を目標値の変化どおりに追従することであり，他の一つは，同じ例で，負荷や原動機速度が変動しても，制御量である出力端子電圧を一定に保とうとすることである。

前者の機能に着目した制御を**追値制御**（ついち）といい，後者の機能に着目した制御を**定値制御**という。

追値制御で，目標値が時間的に任意に変化するような制御を**追従制御**または**自動追尾**（ついび）という。これに対して，目標値をある定められたプログラムに従って変化させるような制御をフィードバック制御における**プログラム制御**という。p. 174のサーボ機構は，この一例である。

定値制御では，原則として目標値は一定に保たれ，**設定値**（set point）とよばれることが多い。p. 183のプロセス制御は，この一例である。

（2） フィードバック制御系の構成

図2は，フィードバック制御系の構成を一般的に示したものである。

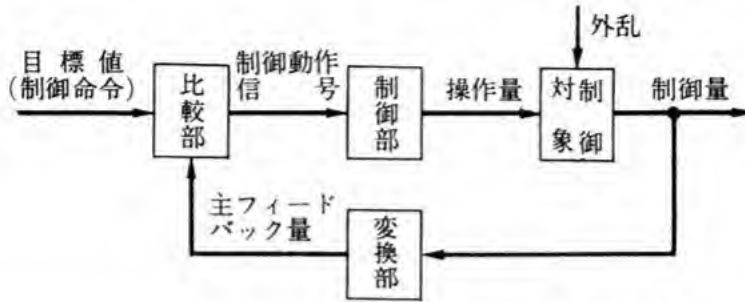



図2 フィードバック制御系の構成

外　乱　　制御量を目標値から外させようとする系の外部からの影響である。図1の系で，原動機速度や負荷の変動は外乱である。

主フィードバック量　　目標値と比較するために，比較部へフィードバックさせる量である。

制御動作信号　　目標値と主フィードバック量との差であって，**制御偏差** または **誤差** ということもある。例えば，図1の系では，v_r と v_c の差すなわち v_e が制御動作信号である。

変換部　　制御量を目標値と比較するために，目標値と同種類の物理量（主フィードバック量）に変換する部分である。図1では，制御量・目標値ともに直流電圧であるので，変換部を必要としないが一般には必要である。

比較部　　目標値と主フィードバック量とから制御動作信号を求める部分である。図1では，目標値・主フィードバック量が電圧であるので，主フィードバック量の符号を逆向きにしたものと，目標値とを直列接続することによって，比較部の働きをしている。

変換部と比較部を一括して，誤差検出を行うという意味から，**検出部** ということがある。

制御部　　制御動作信号を増幅して制御対象に加える部分で，**制御装置** または **増幅部** ということもある。図1では，増幅器がこれに

相当する。また，図1の発電機の界磁電流のように，制御部の出力信号を**操作量**という。

問 1. 定値制御と追値制御とはどのように違うか。

問 2. p.113の図2のシーケンス制御系にも，検出部を通して制御対象から命令処理部へフィードバックがある。このフィードバックと，フィードバック制御でのフィードバックとの役割は，どのように違うか。

2. 伝達関数とブロック線図

(1) 伝達関数とブロック線図

ある要素の入力信号と出力信号との間の関係を，簡単に表すものとして，伝達関数がある。

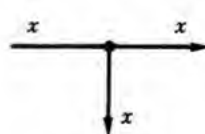
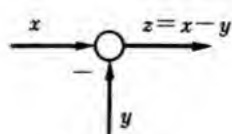
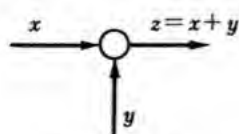
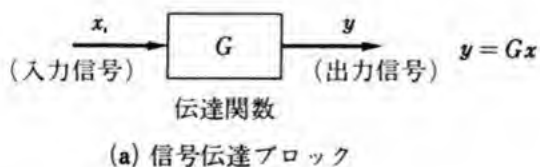
例えば，増幅度が A の増幅器の入力電圧を x，出力電圧を y とすると，これらの間には，次のような関係がある。

$$y=Ax \qquad (7)$$

上式は，x という入力信号が増幅器を通して A 倍されて伝達され，出力信号 y となったことを表す。このことを一般に拡張して，ある

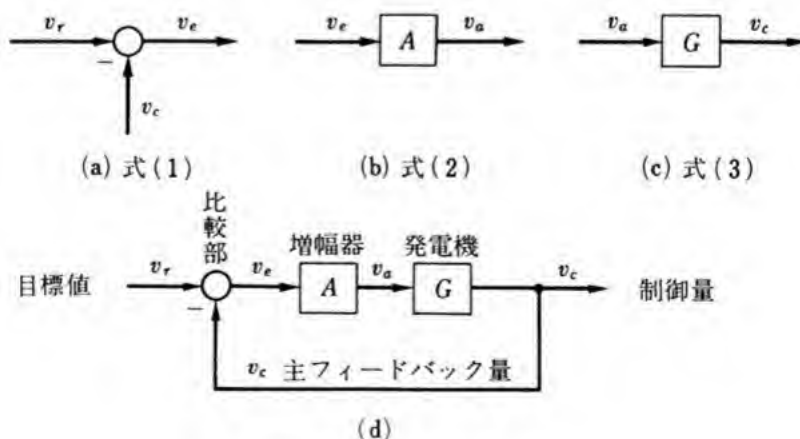
図 3　ブロック線図の基本図記号

(a) 信号伝達ブロック

(b) 加算点　(c) 減算点　(d) 分岐点

要素の入力信号 x, 出力信号 y を後に学ぶ演算子 G を用いて $y=Gx$ と表すとき, G を**伝達関数** (transfer function) という。G が定数の場合をとくに**伝達比**, あるいは**比例ゲイン**といい, このような伝達関数をもつ要素を**比例要素**という。これは, 伝達関数のうち, 最も簡単なものである。

伝達関数を通しての信号の伝達, さらに信号の加算・減算・分岐を, 図3の図記号で表現したものを**ブロック線図** (block diagram) という。図(a)でブロックの中に伝達関数が書き込まれ, これに入る矢印の線が入力信号側, 出る矢印の線が出力信号側を表している。また, 図(b)は二つの信号の加算を表す加算点, 図(c)は一つの信号から他の信号を減じた減算を表す減算点の図記号である。図(d)は一つの信号を2方向に伝える分岐点を表している。

これらの図記号を用いて, 図1のブロック線図を求めると, まず, 式(1)~(3)はそれぞれ図4(a)~(c)のように表されるので, これらを同じ信号に注目して接続すると, 図(d)のブロック線図が得られる。

図4 直流発電機のフィードバック制御のブロック線図

(2) 微分演算子とその性質

関数を元とする集合を定義域・値域とする写像を**演算子**，または**作用素**という。いま，変数 t について無限回微分可能な関数ばかりからなる集合を考える。

時間 t の関数 x, y があり，y が x の導関数であるとき，すなわち

$$y=\frac{dx}{dt} \tag{8}$$

の関係があるとき，$\frac{d}{dt}$ を s で表し，式 (8) を

$$y=sx \tag{9}$$

と書く。このような s を**微分演算子**とよんでいる。

(a) 線　形　性

関数 x_1, x_2 が微分可能であれば，

$$\frac{d(x_1+x_2)}{dt}=\frac{dx_1}{dt}+\frac{dx_2}{dt} \tag{10}$$

であるが，これを s を用いて書き直すと，次式が得られる。

$$s(x_1+x_2)=sx_1+sx_2 \tag{11}$$

次に a が定数の場合，

$$\frac{d}{dt}(ax)=a\frac{dx}{dt} \tag{12}$$

がなりたつ。これを s を用いて書き直すと，

$$s(ax)=asx \tag{13}$$

すなわち，a 倍されたものを t で微分することは，x に sa，または as を掛ける形で実現できる。これで定数 a と演算子 s との積も意味をもったことになる。式 (11) と式 (13) を満たす演算子を**線形的**であるという。この性質によって，重ね合わせの理がなりたつ。

(b) 高次導関数

x と z の間に,

$$z=\frac{d^2x}{dt^2}=\frac{d}{dt}\left(\frac{dx}{dt}\right) \tag{14}$$

の関係があるとする。これは s を用いて, $z=s(sx)$ と書けるから,

5

$$z=s(sx)=s^2x \tag{15}$$

と表す。つまり, 演算子 s と s の積が意味をもつことになる。同様にして s^n も意味をもち, t についての n 回の微分 $\frac{d^n}{dt^n}$ を表す。

この演算子の累乗は, 指数法則

10

$$\left.\begin{aligned} s^m s^n &= s^{m+n} \\ (s^m)^n &= s^{mn} \end{aligned}\right\} \tag{16}$$

を満足する。ここで, m, n は自然数である。

(c) 演算子 $f(s)$

次に, この考え方を拡張して, 微分式 $a\frac{dx}{dt}+bx$ を表すのに $(as+b)x$ と書く。すなわち,

15

$$(as+b)x\equiv a\frac{dx}{dt}+bx \tag{17}$$

として, 新しい形の演算子 $as+b$ を定義する。さらに 2 個の演算子 a_1s+b_1, a_2s+b_2 の積を次のように定義する。すなわち, まず演算子 a_2s+b_2 を施し, その結果に a_1s+b_1 を施すという演算を, これら二つの演算子の積とよんで, $(a_1s+b_1)(a_2s+b_2)$ と書くことにする。定義により,

20

$$\begin{aligned}(a_1s+b_1)(a_2s+b_2)x &= (a_1s+b_1)(a_2sx+b_2x) \\ &= a_1s(a_2sx+b_2x)+b_1(a_2sx+b_2x) \\ &= a_1a_2s^2x+(a_1b_2+b_1a_2)sx+b_1b_2x\end{aligned}$$

だから, 演算子の積はその順序に関係しない。

n 個の演算子の積の意味も明らかであろう。一般にこれらの式の右辺は，

$$c_0s^nx+c_1s^{n-1}x+\cdots\cdots+c_nx$$

という形をしているが，これを

$$(c_0s^n+c_1s^{n-1}+\cdots\cdots+c_n)x \qquad (18)$$

と書いて，$c_0s^n+c_1s^{n-1}+\cdots\cdots+c_n$ を一つの演算子と定義する。これは，形式的には s の多項式だから，

$$f(s)\equiv c_0s^n+c_1s^{n-1}+\cdots\cdots+c_n$$

とおいて，式 (18) を $f(s)x$ と書く。つまり，s の多項式も一つの演算子である。

$f_1(s)$ および $f_2(s)$ を s の多項式演算子とすれば，$f_1(s)y+f_2(s)y$ という結果を与える演算子を $f_1(s)+f_2(s)$ と書いて，これを演算子 $f_1(s)$ と $f_2(s)$ の和と定義することにする。

積 $f_1(s)f_2(s)$ もまったく同様に定義される。このように定義すれば，和，積がその順序に関係しないことは，明らかであろう。

(d) 積 分 演 算

まず，$sy=x$ すなわち $\dfrac{dy}{dt}=x$ ならば，$y=\int xdt$ であるが，これを記号で

$$y=\frac{1}{s}x\equiv s^{-1}x$$

と書くことにする。つまり $\dfrac{1}{s}$ は，t で積分するという演算子である。その際，積分定数を無視する。そうすれば，明らかに

$$s\cdot s^{-1}=s^{-1}\cdot s=1$$

がなりたつ。n 回続けて積分することを $s^{-n}=\dfrac{1}{s^n}$ と表せば，式 (16) は任意の整数 m, n についてなりたつ。もちろん $s^0=1$ と定義する。

(e) **演算子 $\dfrac{1}{f(s)}$**

もし，

$$(as+b)y=x \qquad (19)$$

であれば，これを記号的に

5

$$y=\frac{1}{as+b}x$$

と書くことにする。演算子 $\dfrac{1}{as+b}$ の具体的な意味は，微分方程式

$$a\frac{dy}{dt}+by=x$$

を解けば明らかである。すなわち，

$$\frac{1}{as+b}x=\frac{1}{a}\varepsilon^{-\frac{b}{a}t}\left\{\int x\varepsilon^{\frac{b}{a}t}dt\right\}$$

10 である。もちろん積分定数を無視している。

さらに，$f(s)y=x$ なら，これを記号的に $y=\dfrac{1}{f(s)}x$ と書くことにする。また，$f_1(s)y=f_2(s)x$ なら，

$$y=\frac{f_2(s)}{f_1(s)}x=g(s)x$$

と書いて，s の有理式の演算子が定義できる。

15 以上の事がらからわかるように，微分演算子 s を生成元とする演算子は，s の多項式または有理式で表され，その性質は，代数学の多項式または有理式の性質とまったく変わるところがないのである。

次に学ぶ微分作用・積分作用を含む要素の伝達関数は，演算子を用いて定義されるのである。

（3） 微分作用・積分作用を含む要素の伝達関数

図5(a)のインダクタンス L をもつコイルに流れる電流 i に対する電圧 v は，次式で表される。

$$v=L\frac{di}{dt} \tag{20}$$

この式を微分演算子 s を用いて表すと，次のように書ける。　5

$$v=Lsi \tag{21}$$

i を入力信号，v を出力信号とすると，図3(a)との対応から，コイルの i から v までの伝達関数 G は，式(21)から演算子 Ls に等しい。すなわち，

$$G=Ls \tag{22}$$　10

このような伝達関数をもつ要素を**微分要素**という。

図5(b)の静電容量 C をもつコンデンサに流れる電流 i に対する電圧 v は，次式で表される。

$$v=\frac{1}{C}\int idt \tag{23}$$

s を用いて表すと，次のように書ける。　15

$$v=\frac{1}{Cs}i \tag{24}$$

したがって，コンデンサの i から v までの伝達関数 G は，式(24)から演算子 $\frac{1}{Cs}$ に等しい。すなわち，

図5　コイルとコンデンサにおける電流と電圧との関係

(a) $v=L\frac{di}{dt}$　　(b) $v=\frac{1}{C}\int idt$

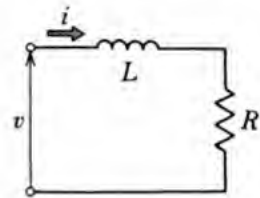
$$G=\frac{1}{Cs} \tag{25}$$

このような伝達関数をもつ要素を**積分要素**という。

以上のように，要素に微分作用・積分作用がある場合，一般に，伝達関数は式 (22)，式 (25)，または後に学ぶ式 (33) のように，s の有理関数となる。そこで，伝達関数を $G(s)$ で表す。

〔**例**〕 図6に示す回路があり，電圧 v と電流 i との間には，次の関係がある。

図 6

$$v=L\frac{di}{dt}+Ri$$

この式を s を用いて表すと，次のように書ける。

$$v=Lsi+Ri=(Ls+R)i$$

i から v までの伝達関数は $Ls+R$ であるが，いま v を入力電圧とし，i を結果としての出力電流とすれば，

$$i=\frac{1}{Ls+R}v$$

となり，この場合の伝達関数 G は，次のようになる。

$$G=\frac{1}{Ls+R}$$

(a)

(b)

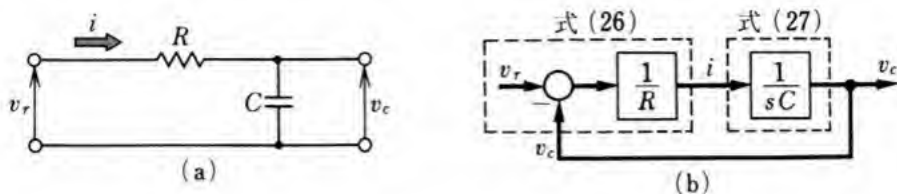



図 7　*CR* 回路のブロック線図

(4)　ブロック線図の例

図 7 (a) の CR 回路のブロック線図を求める。 左側の端子に電圧 v_r が加えられ，右端に電圧 v_c が現れると，電流 i は，

$$i=\frac{1}{R}(v_r-v_c) \tag{26}$$

また，この電流がコンデンサに流れたとき，コンデンサの両端の電圧は，式 (24) で求めたように，次のようになる。

$$v_c=\frac{1}{sC}i \tag{27}$$

これをブロック線図で表せば，図 7 (b) となる。

問 3.　図 8 の RL 回路について，ブロック線図を求めよ。

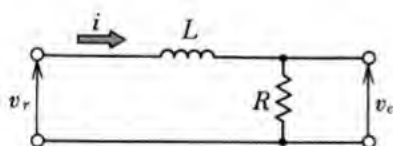



図 8

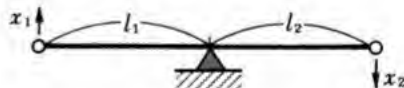



図 9

問 4.　図 9 の てこで，左端の変位 x_1 に対する右端の変位 x_2 を示すブロック線図を求めよ。

問 5.　図 5 (a) のコイルの代わりに抵抗値 R の抵抗を入れると，式 (22) およびブロック線図はどうなるか。

3. ブロック線図の等価変換

(1) 等価変換の法則

ブロック線図は，伝達関数を図形化したものだから，伝達関数と同様に置き換えができる。このときの法則について調べる。これらの性質は，作図上の約束を除き，すべて伝達関数が演算子であることから導かれる。

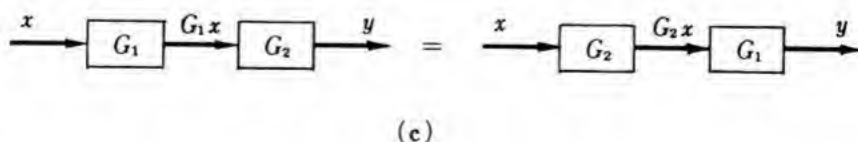
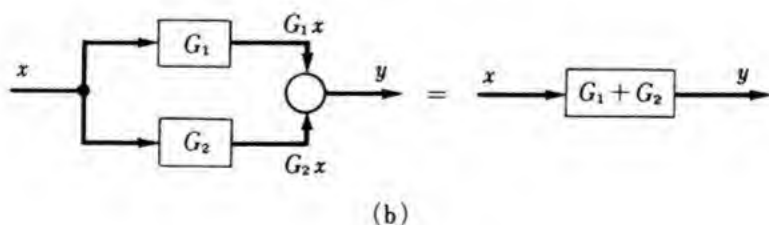
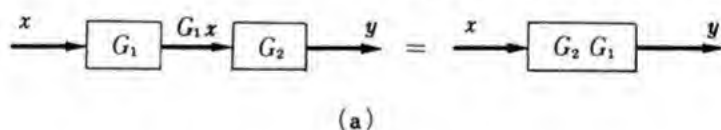
1) 図10(a)のように，2個のブロックが直列に接続されている場合，それぞれの伝達関数の積を伝達関数とする一つのブロックにまとめることができる。演算子の積は，また一つの演算子だからである。

2) 図(b)のように，2個のブロックが並列に接続されている場合，それぞれの伝達関数の和を伝達関数とする一つのブロックにまとめることができる。演算子の和は，演算子だからである。こ

図 10 ブロック線図の等価変換（1）

(a)

(b)

(c)

れは次の数式に相当する。

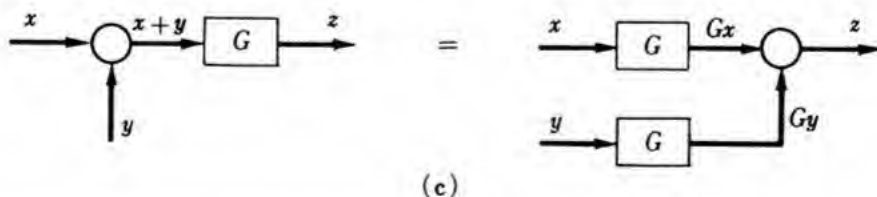
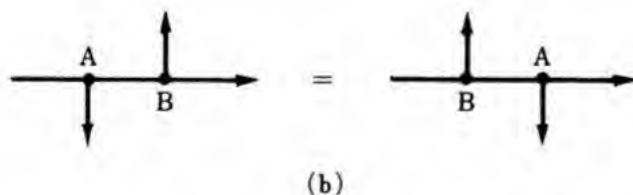
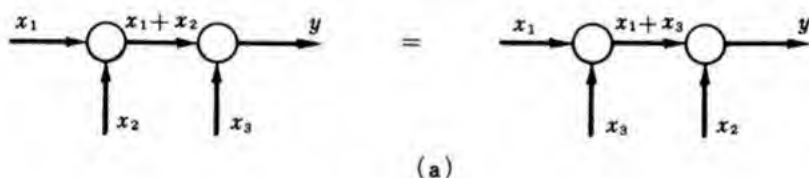
$$y = G_1 x + G_2 x = (G_1 + G_2) x \tag{28}$$

3)　図(c)のように，入力信号と出力信号との間にいくつかの伝達関数がある場合，その順序を任意に交換できる。sの多項式演算子または有理式演算子は，積に関して可換性をもつからである*。以後，演算子というのは，sの多項式演算子または有理式演算子のことをさすものとする。この等価変換は，次の数式に相当する。

$$y = G_2 G_1 x = G_1 G_2 x \tag{29}$$

4)　図11(a)のように，隣り合った加算点はその位置を交換してもよい。和に関する演算子の可換性による。

5)　図(b)のように，隣り合った分岐点の順序は交換してもよい。

図 11　ブロック線図の等価変換（2）

*　一般の演算子は，可換性をもつとは限らない。

これは作図上の約束である。

6) 図(c)のように，入力側の加算点をブロックの出力側に移動した場合の等価変換は，次の数式に相当する。これは演算子 G の線形性である。

$$z=G(x+y)=Gx+Gy \tag{30}$$

7) 出力側の分岐点を入力側に移動した場合は，図12(a)のようになる。これは作図上の約束である。

8) フィードバックループを含むブロック線図は，図(b)のように一つのブロックに等価変換できる。なぜならば，図(b)の左側のブロック線図において，y が H を通ったときの信号 Hy を x から差し引き，さらに G を通すと再び y になるから，

$$G(x-Hy)=y \tag{31}$$

したがって，

$$y=\frac{G}{1+GH}x \tag{32}$$

すなわち，式 (32) をブロック線図で表すと，図(b)の右側のようになる。

図 12 ブロック線図の等価変換（3）

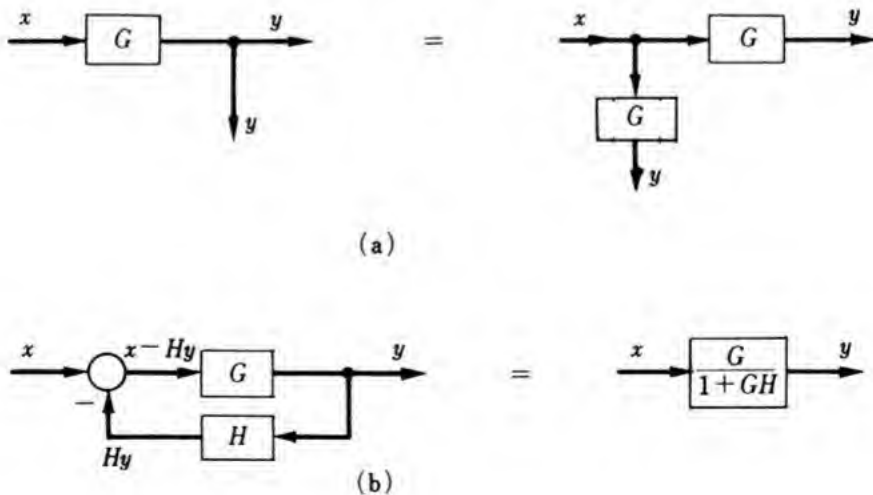
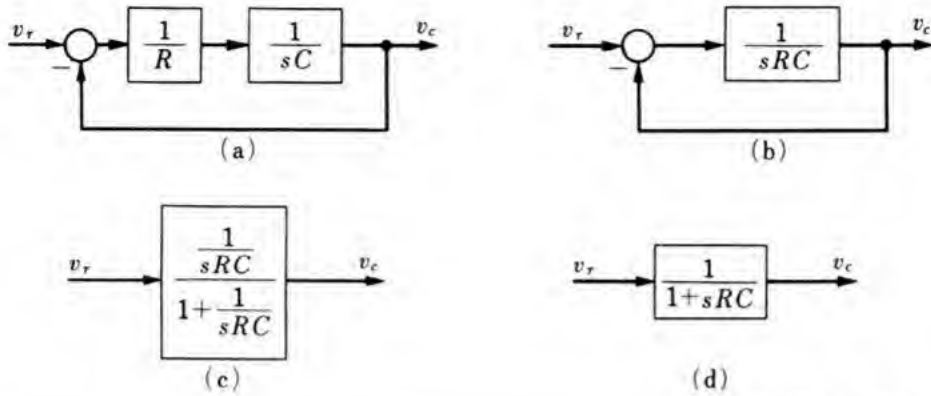



図 13　等価変換の例

(2)　等価変換の例

図7(b)の CR 回路のブロック線図を等価変換して，v_r から v_c までの伝達関数を求める変換の手順を図13に示す。

図(a)から図(b)は等価変換の法則1)が，図(b)から図(c)は8)が用いられている。図(d)はブロック内の伝達関数を整理したものである。以上のようにして，この CR 回路の伝達関数 $G(s)$ は，次のようになる。

$$G(s)=\frac{1}{1+sRC} \tag{33}$$

問 6.　図14のブロック線図を簡単化し，r, c 間の伝達関数を求めよ。

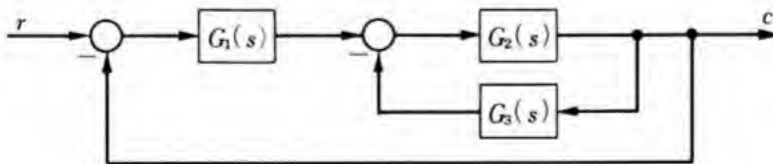



図 14

問 7.　問3で求めたブロック線図を簡単化し，v_r から v_c までの伝達関数を求めよ。

4. 伝達関数の基本形とブロック線図

(1) 伝達関数の基本形

伝達関数の形にはいろいろなものがあるが，その基本的なものとして，すでに学んだ比例要素・積分要素・微分要素のほかに，次に学ぶ一次遅れ要素・二次遅れ要素の伝達関数がある。

一次遅れ要素　図15のように，分子は定数で，分母が s について一次式になるような伝達関数を**一次遅れ伝達関数**といい，このような伝達関数をもつ要素を**一次遅れ要素**という。ここで，K を**ゲイン定数**，T を**時定数**という。

$$v_r \rightarrow \boxed{\frac{K}{1+sT}} \rightarrow v_c$$

図15　一次遅れ要素の伝達関数

$$v_r \rightarrow \boxed{\frac{K\omega_n^2}{s^2+2\zeta\omega_n s+\omega_n^2}} \rightarrow v_c$$

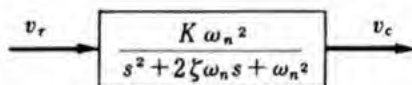
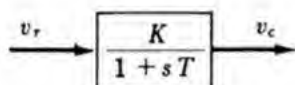



図16　二次遅れ要素の伝達関数

例えば，前ページで求めた伝達関数の式(33)からわかるように，図7(a)の CR 回路は，一次遅れ要素ということができる。この場合，ゲイン定数 K は1，時定数 T は RC である。

二次遅れ要素　図16のように，分母が s について二次式の形の伝達関数を**二次遅れ伝達関数**といい，このような伝達関数をもつ要素を**二次遅れ要素**という。ここで K はゲイン定数であるが，ω_n を**固有角周波数**，ζ を**減衰係数**という。

(2) 伝達関数とブロック線図の実例（直流発電機の例）

図17の直流発電機を例にして，制御機器の伝達関数とブロック線図を求めてみる。

まず，界磁回路で界磁電流 i_f は，

$$i_f=\frac{1}{R_f+sL_f}v_a \tag{34}$$

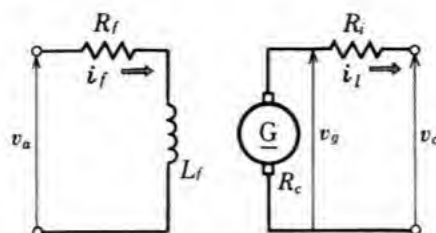



図 17　直流発電機の構成

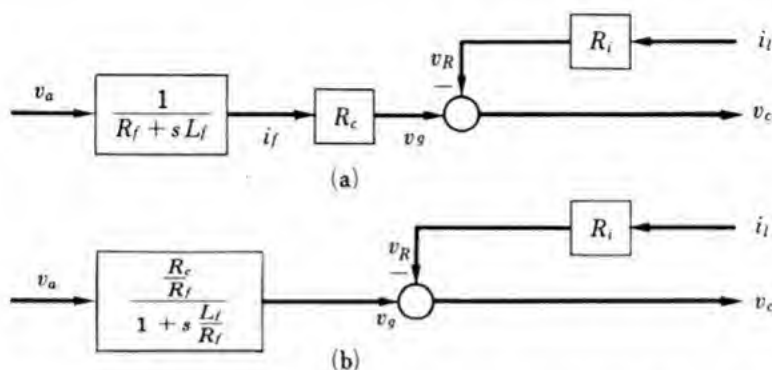



図 18　図17の直流発電機のブロック線図

また，i_f と発生起電力 v_g との関係は，次のようになる。

$$v_g = R_c i_f \tag{35}$$

ここで，R_c は発電定数で，その単位は抵抗と同様 Ω である。

出力端子電圧 v_c は，負荷電流 i_l が流れているとすると，v_g, i_l から，

$$v_c = v_g - R_i i_l \tag{36}$$

となる。ただし，R_i は直流発電機の電機子抵抗とする。式 (34), (35), (36) をブロック線図で表すと，図 18 (a) となる。図(b)は，これを等価変換して整理したものである。

問 8.　抵抗の単位を〔Ω〕，静電容量の単位を〔F〕とすると，時定数 T の単位は〔s〕となる。図 7 (a) の回路で，$R=100$〔kΩ〕，$C=5$〔μF〕とすると，時定数 T は何秒か。

問 9.　図 17 の直流発電機の出力端子に，抵抗値 R_l の負荷を接続すると，ブロック線図はどのようになるか。また，そのときの v_a から v_c までの伝達関数を求めよ。

5. フィードバック制御系のブロック線図

(1) フィードバック制御系のブロック線図の例

一例として，p. 136 で学んだ直流発電機の自動電圧制御系のブロック線図を求めてみよう。

図 19 において，直流発電機の部分のブロック線図は，図 18 (b) に示したとおりである。したがって，増幅部および比較部のブロック線図をつけ加えることにより，図 20 (a) が得られ，図 (b) のように整理できる。

図 19 直流発電機の自動電圧制御系

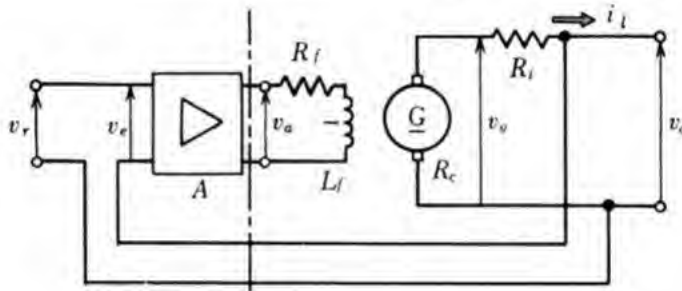



図 20 図 19 の自動電圧調整系のブロック線図

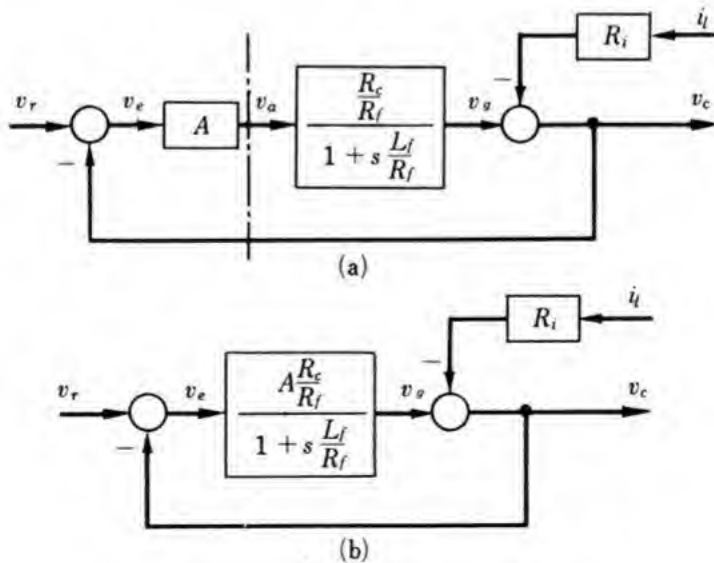
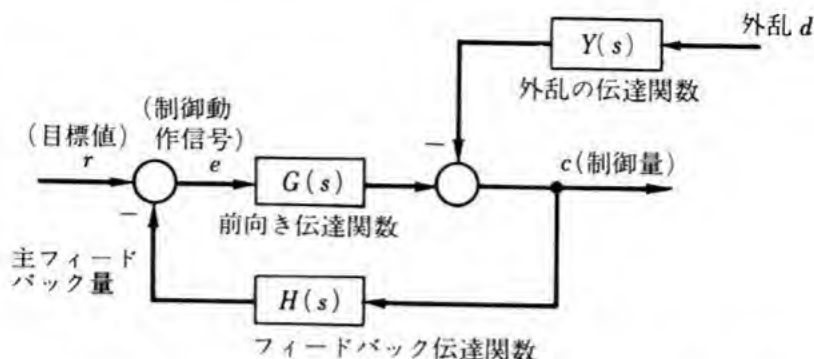



図 21 フィードバック制御系のブロック線図の一般形

(2) フィードバック制御系のブロック線図の一般形

図21は，図20のようなフィードバック制御系のブロック線図の一般形を示したもので，図21の $G(s)$ は**前向き伝達関数** (forward transfer function) といい，図20(b) では $G(s)$ は，次のとおりである。

$$G(s)=\frac{A\dfrac{R_e}{R_f}}{1+s\dfrac{L_f}{R_f}} \tag{37}$$

$G(s)$ は，ふつう，制御部の伝達関数と，制御対象の伝達関数との積に等しい。

$H(s)$ を**フィードバック伝達関数** (feedback transfer function) といい，一般には定数であることが多い。図20(b) では $H(s)=1$ である。

$Y(s)$ を**外乱の伝達関数**という。図20(b) では $Y(s)=R_i$ である。

(3) フィードバック制御系の伝達関数

図21より，目標値 r から制御量 c までの伝達関数 $M(s)$ を求めると，次のようになる。すなわち，同図で外乱を0とすると，$G(s)$ のブロックの次にある加算点へ $Y(s)$ のブロックから入る信号は0となるので，この場合のブロック線図は，図12(b) と一致する。したがって，$M(s)$ は，等価変換の法則8) より次のようになる。

$$M(s)=\frac{G(s)}{1+G(s)H(s)} \tag{38}$$

この伝達関数を**閉ループ伝達関数**（closed loop transfer function）とよんでいる。

一般に，式（38）を整理して，その分母が s について一次多項式になる場合，そのフィードバック制御系を**一次系**，二次多項式になる場合を**二次系**とよんでいる。また，三次多項式以上の系を**高次系**とよぶ。

例1　一次系の例

図20(b)の系は，一次系である。図で i_l を0とし，また，図22(a)のような置き換えを行うと，この閉ループ伝達関数 $M(s)$ は，式（38）から，

$$M(s)=\frac{\dfrac{K_0}{1+sT_0}}{1+\dfrac{K_0}{1+sT_0}} \tag{39}$$

$$=\frac{K_0}{1+K_0+sT_0} \tag{40}$$

$$\therefore\quad M(s)=\frac{\dfrac{K_0}{1+K_0}}{1+\dfrac{sT_0}{1+K_0}} \tag{41}$$

となり，図(b)のように等価変換されたことになる。式(41)で，

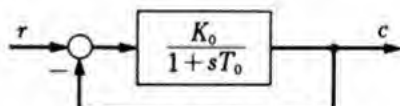
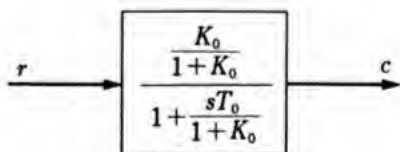
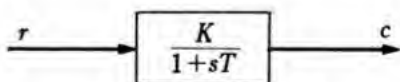
$$K=\frac{K_0}{1+K_0},\quad T=\frac{T_0}{1+K_0} \tag{42}$$

図 22　一次系のブロック線図

(a)

r　$\dfrac{\frac{K_0}{1+K_0}}{1+\frac{sT_0}{1+K_0}}$　c

(b)

r　$\dfrac{K}{1+sT}$　c

$K=\frac{K_0}{1+K_0},\ T=\frac{T_0}{1+K_0}$

(c)

とおけば,

$$M(s)=\frac{K}{1+sT} \tag{43}$$

となり，一次遅れ伝達関数となる。

例2　二次系の例

図20(a)で，増幅部の伝達関係が一次遅れ伝達関数

$$\frac{A}{1+sT_a} \tag{44}$$

であったとすると，この系は，図23(a)のブロック線図で示すように，二次系となる。

この系の閉ループ伝達関数 $M(s)$ は,

$$M(s)=\frac{\dfrac{K_0}{(1+sT_a)(1+sT_0)}}{1+\dfrac{K_0}{(1+sT_a)(1+sT_0)}} \tag{45}$$

$$=\frac{K_0}{1+K_0+s(T_a+T_0)+s^2T_aT_0} \tag{46}$$

$$\therefore\quad M(s)=\frac{\dfrac{K_0}{T_aT_0}}{s^2+s\left(\dfrac{1}{T_a}+\dfrac{1}{T_0}\right)+\dfrac{1+K_0}{T_aT_0}} \tag{47}$$

となり，図23(b)のように等価変換されたことになる。上式で,

$$\frac{1+K_0}{T_aT_0}=\omega_n^2,\quad \frac{K_0}{T_aT_0}=K\omega_n^2,\quad \left(\frac{1}{T_a}+\frac{1}{T_0}\right)=2\zeta\omega_n \tag{48}$$

または，式(48)から,

$$\left.\begin{aligned}\omega_n&=\sqrt{\frac{1+K_0}{T_aT_0}},\\K&=\frac{K_0}{1+K_0},\\\zeta&=\frac{T_a+T_0}{2\sqrt{T_aT_0(1+K_0)}}\end{aligned}\right\} \tag{49}$$

とおくと，

$$M(s)=\frac{K\omega_n^2}{s^2+2\zeta\omega_n s+\omega_n^2} \tag{50}$$

となる。これを図23(c)に示す。このように，ふつう二次系の閉ループ伝達関数の分母は，ω_n, ζ を用いて表されることが多い。

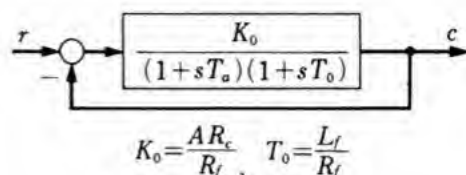
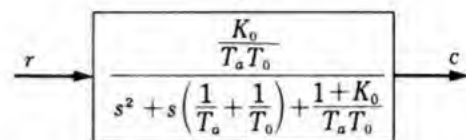
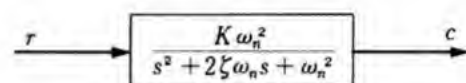
r, c, $\frac{K_0}{(1+sT_a)(1+sT_0)}$

$K_0=\frac{AR_c}{R_f}$, $T_0=\frac{L_f}{R_f}$

(a)

r, c, $\dfrac{\frac{K_0}{T_aT_0}}{s^2+s\left(\frac{1}{T_a}+\frac{1}{T_0}\right)+\frac{1+K_0}{T_aT_0}}$

(b)

r, c, $\frac{K\omega_n^2}{s^2+2\zeta\omega_n s+\omega_n^2}$

$K=\frac{K_0}{1+K_0}$, $\omega_n^2=\frac{1+K_0}{T_aT_0}$,

$\zeta=\frac{T_a+T_0}{2\sqrt{T_aT_0(1+K_0)}}$

(c)

図 23　二次系のブロック線図

問 10． 前向き伝達関数が $\frac{10}{s(1+2s)}$，フィードバック伝達関数が2であるようなフィードバック制御系のブロック線図をかけ。

問 11． 前問の系の閉ループ伝達関数を求めよ。また，このときの ω_n, ζ の値はいくらか。

6．フィードバック制御系の特性とその改善

(1) 特性の種類

一般に，ある装置がどのような性質をもち，どのような動作をするかを示すものを**特性** (characteristic) といい，ある要素または系に入力信号が加わったとき，入力信号の変化に対して，出力信号が時間的にどのように変化するかを示す特性を**応答** (response) という。

フィードバック制御系に，外部から加わる信号としては，目標値と外乱がある。したがって，フィードバック制御系の主な特性または応答としては，目標値と制御量との間の特性（目標値に対する制御量の応答）と，外乱と制御量との間の特性（外乱に対する制御量の応答）との二つがある。

特性を調べるために，外部から加える信号には，いろいろなものがある。例えば，図24(a)のフィードバック制御系の目標値を，図(b)のように急変したとすると，変化直後は，制御量が目標値の変化に追い着くことができず，例えば図(c)のような経過をたどり，やがて目標値に近い一定の値になる。

このように，ある時刻，例えば $t=0$ までは0，それ以後は一定値に急変する信号を**ステップ信号**，これに対する応答を**ステップ応答**とよぶ。とくに，変化の大きさが単位量（電圧であれば1〔V〕）であるステップ信号を**単位ステップ信号**といい，$1(t)$ または $u(t)$ で表す。

また，入力信号として正弦波信号を加えたときの応答も，特性の評価に有効である。これについては後に学ぶ。

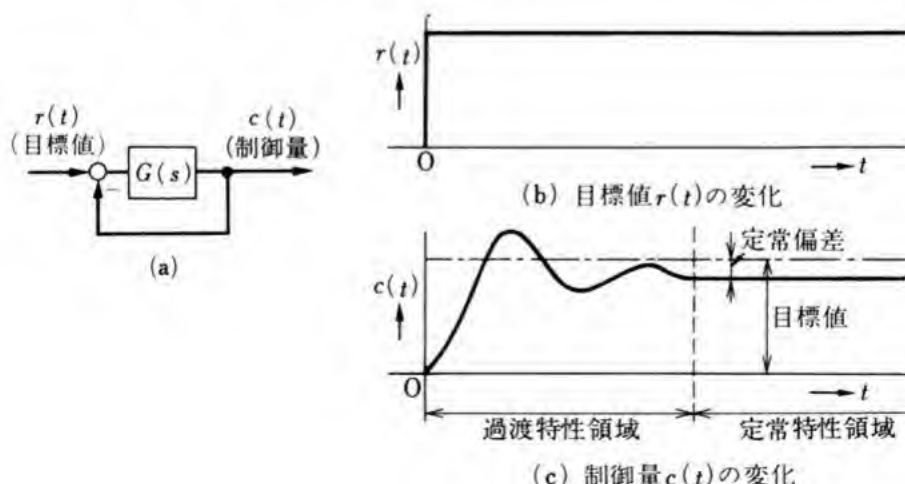
図 24 フィードバック制御系の目標値の変化に対する制御量の変化

(2) ステップ応答の定常特性

図24(c)に示すように，応答が，入力信号に対応する一定の値に落ち着くまでの特性を**過渡特性**，落ち着いてから後の定常状態の特性を**定常特性**という。

5 ステップ応答の定常状態では，系内のすべての信号は変化せず，一定であるから，それらを t で微分すればすべて0となる。したがって，それらに s すなわち演算 $\dfrac{d}{dt}$ を施すことは，0を掛けることと同じであるから，定常特性は，伝達関数の中の s をすべて0とおくことによって求められる。これを p. 157～158 の 例1，例2 に
10 ついて考えてみよう。

例1 一次系の例

式(41)で，から制御量 c は，

$$c=M(s)r=\frac{\dfrac{K_0}{1+K_0}}{1+\dfrac{sT_0}{1+K_0}}r \qquad (51)$$

上式で，定常特性を考えるため，$r=R$（一定値），$s=0$，$c=$
15 C（定常値）とすると，

$$C=M(0)R=\frac{K_0}{1+K_0}R \qquad (52)$$

となる。この式からわかるように，一般に，制御量の定常値 C と，一定目標値 R との間には差がある。これを E とすると，式(52)から，

20
$$E=R-C=\frac{1}{1+K_0}R \qquad (53)$$

となる。これを**定常偏差**または**オフセット**（off-set）という。この値は，式（53）からわかるように，K_0 を大きくすればするほど小さくなる。

例2　二次系の例

式（47）の $M(s)$ を用いて，例1と同様な条件と方法で制御量の定常値 C を求めると，

$$C=M(0)R=\frac{\dfrac{K_0}{T_aT_0}}{\dfrac{1+K_0}{T_aT_0}}R=\frac{K_0}{1+K_0}R$$

これは，一次系の場合の式（52）と同じであり，したがって，定常偏差 E も式（53）と同じになる。すなわち，この場合，増幅器の時定数 T_a は，定常偏差に無関係となることがわかる。一般に，時定数は，定常偏差に影響を与えない。

（3）フィードバック制御系のステップ応答と過渡特性

フィードバック制御系のステップ応答としては，ふつうは，（1）特性の種類で例に挙げたように，目標値に対する制御量のステップ応答に，注目することが多い。

ステップ応答は，フィードバック制御系が何次系であるかによって，その形が異なってくる。図25は，その相違を示したもので，図(a)は一次系，図(b)は二次系，図(c)は高次系のステップ応答である。

一次系のステップ応答は，図(a)からわかるように，K の値を超えることはない。また，T を小さくすればするほど早く K の値に近づく。これは式（42）の第2式でわかるように，K_0 をより大きく

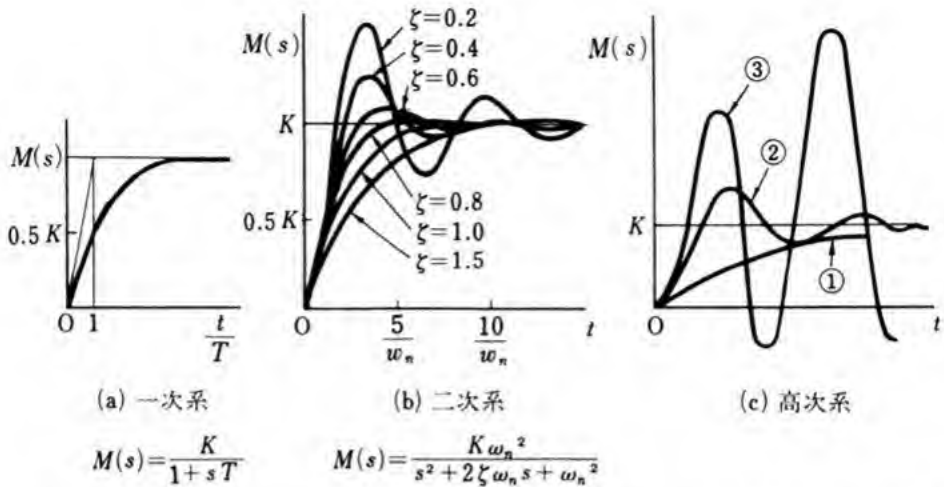



図 25 フィードバック制御系のステップ応答

することに対応する。この場合，定常特性の例で学んだように，定常偏差はより小さくなるので，望ましい結果となる。

一方，二次系のステップ応答は，ω_n の値が大きいほど，その立ち上がりが早いことがわかる。また，ζ が 1.0 より小さくなると振動的になり，とくに 0 に近づくほどより振動的になり，制御系として望ましくない応答となる。したがって，定常偏差の例で学んだように，定常偏差を小さくするために K_0 をより大きくすると，式 (49) の第 3 式から，ζ の値はより小さくなり，ζ の値が 1 以下となってより振動的になる。このように，二次系では，定常特性を良くするために K_0 を大きくしようとすると，その値は過渡特性の方から制限を受ける。このような傾向は，一次系には見られないものであり，高次系になるほど強くなる。

(4) 正弦波入力信号に対する応答

図 24 (a) で，目標値として正弦波入力信号を加えた場合の応答にも，図 26 に示すように，過渡特性領域と定常特性領域とがある。

定常特性領域での応答は，位相はずれるが入力信号と周波数の等

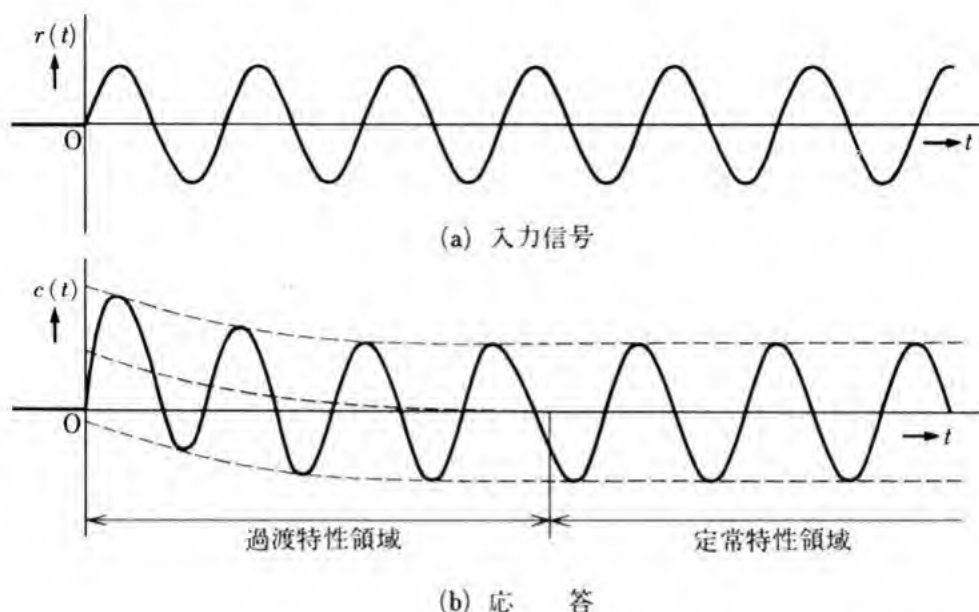



図 26 正弦波入力信号に対する応答

しい正弦波信号となることが知られている。とくに，このような定常特性は，後で学ぶ周波数応答の意味で重要である。定常特性に関しては，「電気基礎」で学んだ記号法に従い，複素量として取り扱うことができる。

例えば，p. 147 の例で，電圧 v を入力信号として加え，過渡状態がじゅうぶんに減衰した後の定常電流 i を考える。ここで，v, i をそれぞれ複素量 $\dot{V}, \dot{I}$ で表すと，次式が得られる。

$$\left.\begin{aligned}\dot{V}&=V(\cos\omega t+j\sin\omega t)&&=V\varepsilon^{j\omega t}\\ \dot{I}&=I\{\cos(\omega t+\theta)+j\sin(\omega t+\theta)\}&&=I\varepsilon^{j(\omega t+\theta)}\end{aligned}\right\}\quad(54)$$

これらの定常値についても，微分方程式

$$L\frac{d\dot{I}}{dt}+R\dot{I}=\dot{V}\quad(55)$$

がなりたつので，式 (55) に式 (54) を代入すると，

$$L\frac{d}{dt}I\varepsilon^{j(\omega t+\theta)}+RI\varepsilon^{j(\omega t+\theta)}=V\varepsilon^{j\omega t}$$

となり，以下変形すると，

$$Lj\omega I\varepsilon^{j(\omega t+\theta)}+RI\varepsilon^{j(\omega t+\theta)}=V\varepsilon^{j\omega t}$$

$$Lj\omega\dot{I}+R\dot{I}=\dot{V} \tag{56}$$

または，
$$\dot{I}=\frac{1}{Lj\omega+R}\dot{V} \tag{57}$$

式 (55) を s を用いて表すと，

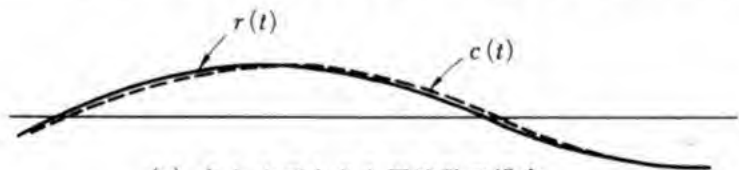
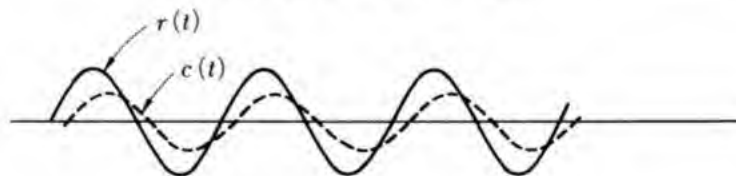
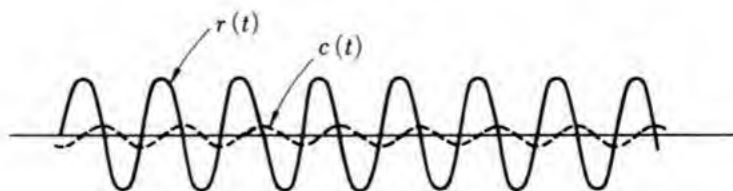
$$Ls\dot{I}+R\dot{I}=\dot{V} \tag{58}$$

または，
$$\dot{I}=\frac{1}{Ls+R}\dot{V} \tag{59}$$

となる。式 (57) と式 (59) を対比することによって，正弦波入力信号の定常特性においては，s は $j\omega$ という具体的な値をとることがわかる。

（5） 周波数応答

いま，図 24 (a) の目標値 $r(t)$ として正弦波信号を加えると，定

図 27　周波数の大きさと $r(t)$ に対する $c(t)$ の追従性の関係

(a) きわめて小さな周波数の場合

(b) 中間の周波数の場合

(c) きわめて大きな周波数の場合

常状態での制御量 $c(t)$ は，図27に示すように，周波数によって，それらの振幅比および位相差が異なる。図(a)では，周波数 ω がじゅうぶん小さいときは，制御量 $c(t)$ は目標値にほぼ追従しているが，周波数が増加するとしだいに遅れはじめ，図(b)のようになり，さらに増加すると，図(c)のようにほとんど追従が不可能となっていることがわかる。このような傾向を特性として示すものに周波数応答がある。

いま，図28に示すように，周波数 ω の正弦波入力信号 $x(t)$ の振幅を A_i，そのときの出力信号 $y(t)$ の振幅を A_0，また入力信号に対する出力信号の位相差を θ で表すと，次のような複素量が得られる。

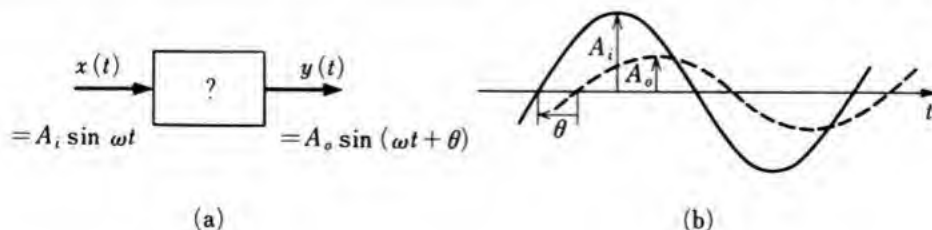
$$\frac{A_0}{A_i}\varepsilon^{j\theta}=\frac{A_0}{A_i}(\cos\theta+j\sin\theta) \qquad (60)$$

ただし，ω は $0\sim\infty$ の値をとる。

この複素量は，ω の値によって変化するが，全周波数に対する変化の様子を**周波数応答**とよぶ。

また，周波数応答は，その要素または系の伝達関数 $G(s)$ があらかじめわかっているとすると，p.165で学んだように，$G(j\omega)$ となる。すなわち，

$$G(j\omega)=\frac{A_0}{A_i}\varepsilon^{j\theta} \qquad (61)$$

図 28 入力信号 $x(t)$ と出力信号 $y(t)$

周波数応答を表すには，周波数を0から∞まで変化させたときの $G(j\omega)=\frac{A_0}{A_i}\varepsilon^{j\theta}$ の値の変化を図で示すことが多い。これには，ベクトル軌跡とボード線図が多く用いられる。

ベクトル軌跡は，ω の変化に対する複素平面上のベクトル $G(j\omega)$ の先端の軌跡である。図29(a)，(b)は，それぞれ一次系および二次系のベクトル軌跡である。

ボード線図（Bode's diagram）は，振幅比 $|G(j\omega)|=\frac{A_0}{A_i}$ と位相角 $\arg G(j\omega)=\theta$ の ω に対する変化の様子をグラフで表したものである。すなわち，横軸を周波数 ω の対数目盛とし，縦軸を

$$g=20\log_{10}|G(j\omega)| \quad [\mathrm{dB}]$$

で定義した**ゲイン** g で表し，ω と g との関係を曲線で示したものを**ゲイン特性曲線**という。また，縦軸を位相角 $\theta=\arg G(j\omega)$ (°) として，ω と θ との関係を曲線で示したものを**位相特性曲線**という。ま

図 29 一次系および二次系のベクトル軌跡

$$G(j\omega)=\frac{K}{1+j\omega T}$$

(a) 一次遅れ要素のベクトル軌跡

$$G(j\omega)=\frac{K\omega_n^2}{(j\omega)^2+2\zeta\omega_n j\omega+\omega_n^2}=\frac{K}{1-\left(\frac{\omega}{\omega_n}\right)^2+j2\zeta\frac{\omega}{\omega_n}}$$

(b) 二次遅れ要素のベクトル軌跡

(a) ゲイン特性曲線

(b) 位相特性曲線

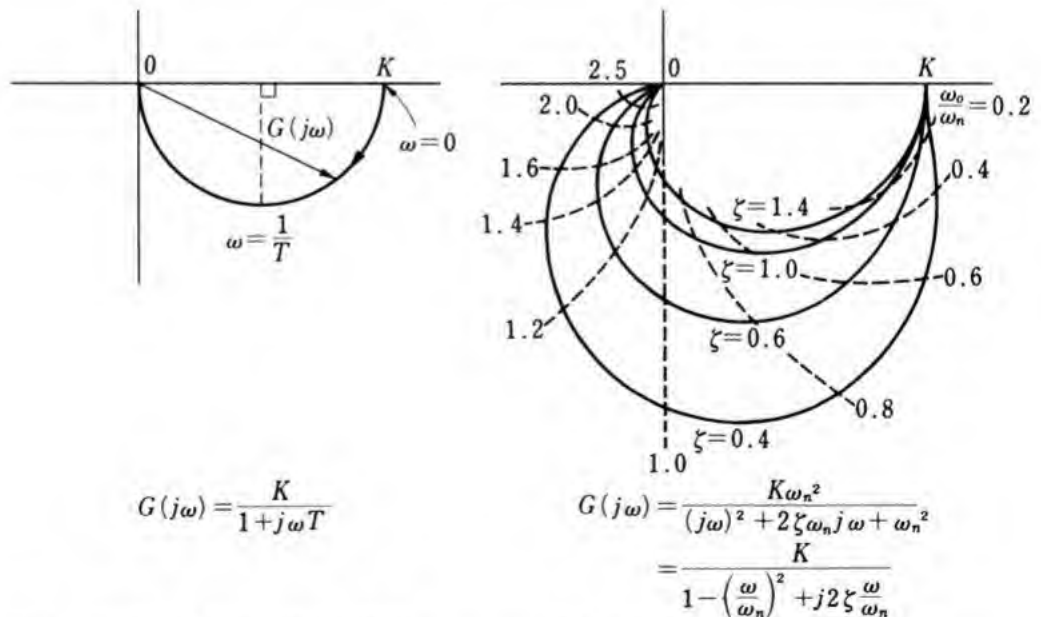
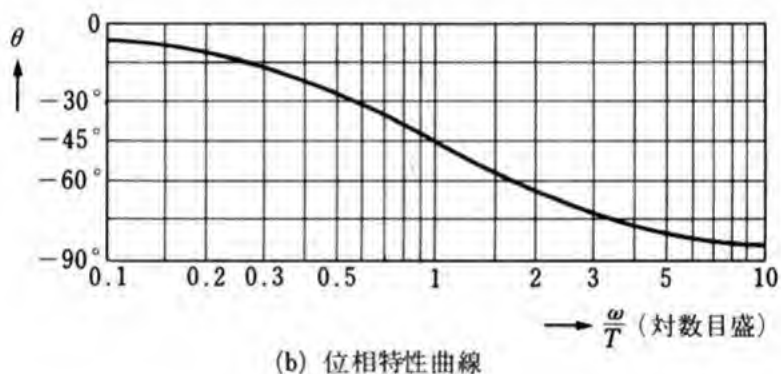
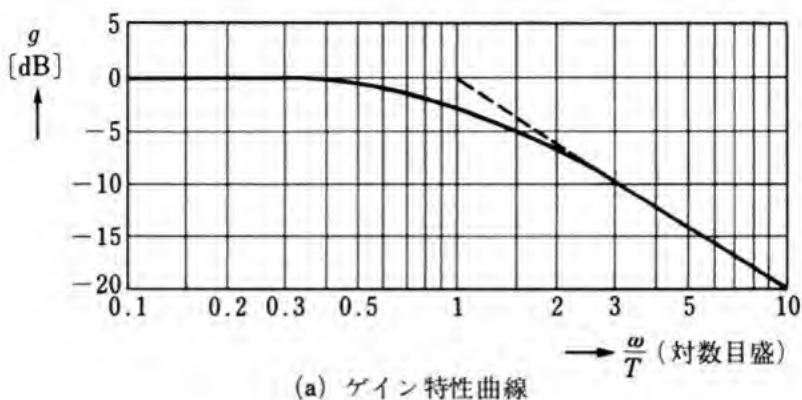



図 30　一次系$\left(G(s)=\frac{1}{1+sT}\right)$のボード線図

た，これらを総称してボード線図という。図30は，一次系のボード線図である。

図30より，周波数ωの増加とともに，ゲインgは限りなく減少し，位相角θは$-90°$に限りなく近づくことがわかる。

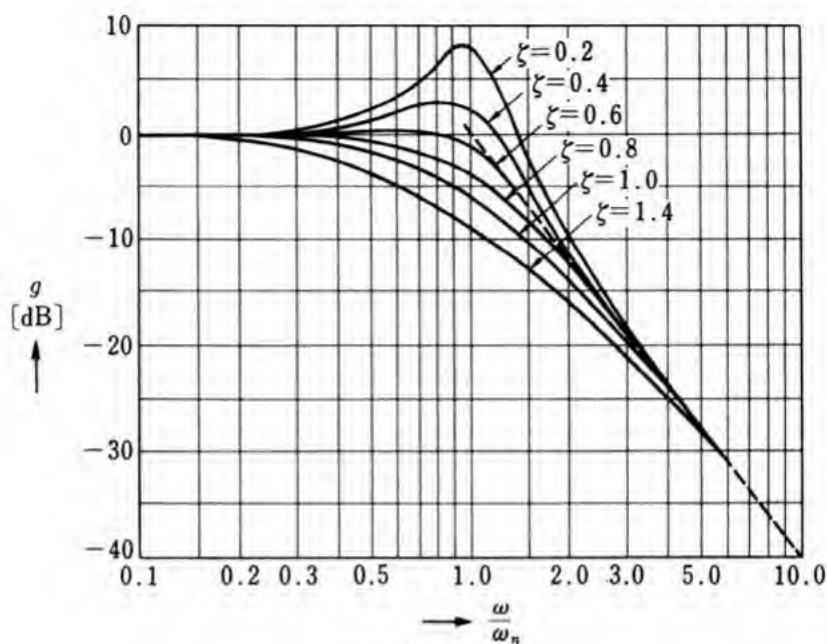
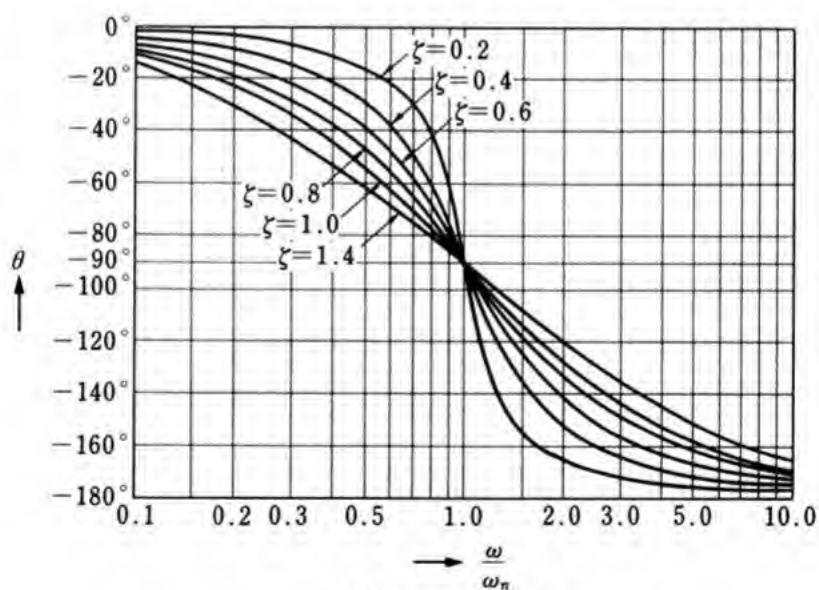
また，図31は，二次系のボード線図である。図(a)のゲイン特性曲線では，周波数がω_n以上になると，ゲインgは周波数ωの増加とともに限りなく減小するが，その速さは一次系の場合の2倍になっている。また，減衰係数ζの値が$\frac{1}{\sqrt{2}}$以下では，周波数がω_nの近くでゲインgの値にピークが生ずる。この性質を**共振特性**とよ

(a) ゲイン特性曲線

(b) 位相特性曲線

図 31 二次系 $\left(G(s)=\frac{\omega_n{}^2}{s^2+2\zeta\omega_n s+\omega_n{}^2}\right)$ **のボード線図**

んでいる。図(b)の位相特性曲線では，位相角 θ は負の値であり，周波数 ω の増加とともに減小し，$-180°$ に限りなく近づくことがわかる。

(6) フィードバック制御系の特性の改善

実用的なフィードバック制御系は，定常偏差が小さく，かつじゅうぶんに早く応答するものでなければならない。ところで二次系の場合，(2) ステップ応答の定常特性で学んだように，定常偏差を小さくするため K_0 を大きくすると，p. 159 の例2の式 (49) に示すように，減衰係数 ζ の値はより小さくなる。とくに，ζ の値が1以下となり0に近づくと，ステップ応答はより振動的となる。また，目標値が絶えず変動するような場合にも，周波数応答における共振特性が著しく，制御量が目標値どおりに変化できない。これらはいずれも望ましいことではない。

また，三次以上の高次系になると，ステップ応答は，図25の①，②のように，二次系の場合と似たものもあるが，ときには③のように，時間とともに応答が発散することがある。このような現象は，二次系の場合と同じように，定常特性を良くしようとすると起こることが多く，このことを**不安定**とよんでいる。このように不安定であったり，逆に一定値に落ち着くまでの時間が長引いたりする，いわゆるわるい応答をもつ場合には，これを改善しなければならない。このように，特性の改善を行うことを**特性補償**という。

特性補償の方法はいろいろあるが，図25の制御部の最初の部分に，**調節要素**を挿入する方法が多く用いられる。調節要素は，図32に示すように，制御動作信号を直接通す部分（比例項），微分して通す部分（微分項），積分して通す部分（積分項）からなりたっており，これらがある一定の比率で加え合わされて，調節信号として出

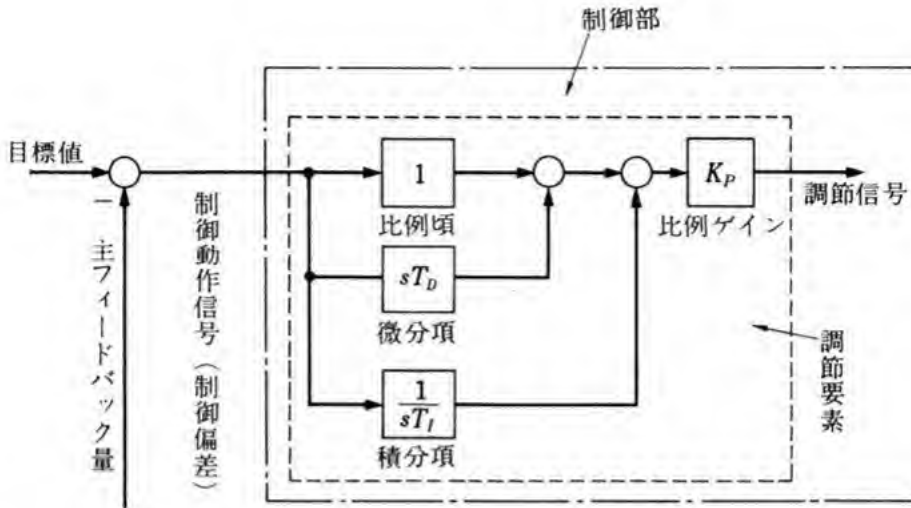



図 32 PID 制御動作の構成

ていくようになっている。これらを人間が手で制御する場合に対応して考えてみる。

まず，比例項だけの場合，目標値と主フィードバック量の間の偏差だけに着目する制御のしかたで，偏差がある間は制御対象に操作を加え続ける。したがって，偏差がなくなったとき，操作を止めることになるが，制御対象はそこですぐに制御量の変化を止めるわけではなく，ちょうど運動における慣性のように，なおも変化し続けるので，偏差が逆向きに現れる。これに従って，逆向きの操作を行うと，再び正の向きの偏差が現れる。このように，すばやく偏差を低減することができない場合が多く，不安定になることもある。

これに対して微分項を付け加えることは，ちょうど熟練した操作者が，偏差の変化速度の大きさと向きを操作の中に加味して，先を見越した操作を行うのと同様であり，すばやく偏差を低減することができる。

積分項を付け加えることは，わずかでも偏差があると，これが完

全に0になるまで操作をし続けるので，精度の高い操作を行うことに相当する。

したがって，これらを適当に組み合わせた調節要素を用いると，じゅうぶんに安定で応答が速く，かつ定常偏差の少ない制御を行わせることができる。

比例項・積分項および微分項の制御動作を，それぞれ**比例（P）動作**（proportional action），**積分（I）動作**（integral action）および**微分（D）動作**（derivative action）とよび，これらの組み合わせで，比例＋微分（PD）動作，比例＋積分（PI）動作のような2項動作，比例＋積分＋微分（PID）動作の3項動作が，制御対象に応じて採用される。

後で学ぶプロセス制御（p. 182）では，このような調節要素が製品として用意されており，これを**調節器**あるいは**調節計**とよんでいる。

問 12. フィードバック制御系の過渡特性と定常特性について説明せよ。
問 13. 一次系・二次系のステップ応答の特徴について述べよ。
問 14. フィードバック制御系の安定・不安定とは何か。
問 15. 微分動作は，特性改善にどのような役割を果たしているか。

問　　題

1. 図33の回路のブロック線図をえがき，v と i の関係を求めよ。

2. 図34のブロック線図を簡略化せよ。

3. 図35の回路のブロック線図をえがけ。

5 **4.** 図36の CR 回路において，時定数を1〔s〕にするためには，C の値をいくらにすればよいか。ただし，$R=500$〔kΩ〕とする。

5. 図17（p.154）において，発電機の負荷として，抵抗 R とインダクタンス L の直列回路が接続されている場合のブロック線図を求めよ。

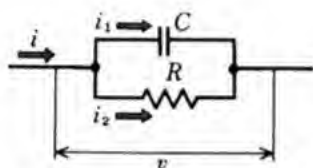
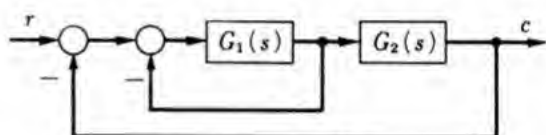
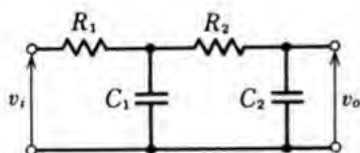
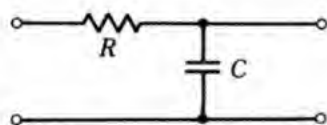
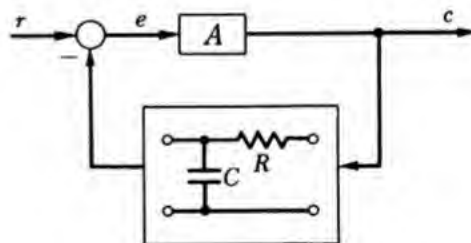



図33　図34　図35　図36　図37

6. 図37のような系において，前向き伝達関数 A（定数）の値が無限大で

10 あるとき，閉ループ伝達関数を求めよ。

7. フィードバック制御系の分母の多項式の次数が多くなるにつれて，ステップ応答にどのような傾向を生ずるか。

8. 特性改善は，どのような場合に行わなければならないか。

自動制御の例

この節のねらい　自動制御は，非常に広い範囲にわたって各産業で用いられ，オートメーションのための基礎技術となっている。ここでは，その代表的なものとして，サーボ機構およびプロセス制御を中心に，どのように利用されているか，それぞれの例について調べる。

1. サーボ機構

(1) サーボ機構

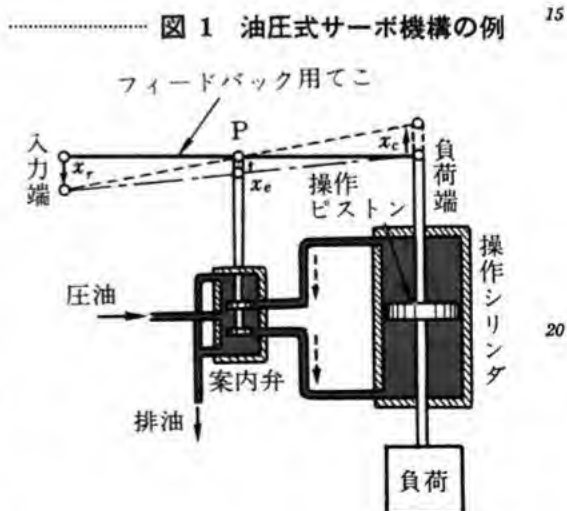
サーボ機構は，制御量が位置・角度であるようなフィードバック制御系で，広い応用範囲をもっている。サーボ機構は，ふつう目標値の変化に追従することを目的としており，いわゆる追値制御系であることが多い。

(2) サーボ機構の種類

サーボ機構には，代表的なものとして，油圧式サーボ機構と電気式サーボ機構とがある。

油圧式サーボ機構　図1は，油圧式サーボ機構の例である。図で，操作ピストンには負荷が結合されており，その位置がこの系の制御量となる。また，フィードバック用てこの左端（入力端）の位置が目標値である。

図1　油圧式サーボ機構の例

この系の目的は，制御量 x_c

を目標値 x_r に比例して追従させることである。ただし，この場合には，目標値 x_r の向きと，制御量 x_c の向きとは反対になっている。

図において，フィードバック用てこの入力端を x_r だけ下向きに動かしたとすると，その瞬間には負荷端は静止していると考えられるから，てこは一点鎖線のように傾き，案内弁のスプールは x_e だけ下向きに動く。そうすると，圧油は操作シリンダに矢印の向きに流入し，操作ピストンを押し上げる。それによってフィードバック用てこの負荷端が上向きに動き，案内弁のスプールを引き上げる。スプールが元の位置にもどると，操作シリンダへの圧油の流れが止まり，操作ピストンが静止する。このとき，てこは図1の破線のような状態になる。

実際には，この動作がかなり速く行われるので，てこの入力端を動かしたとすると，負荷端は，ちょうど点Pがてこの支点であるかのように動く。このとき，負荷端で大きな力が必要な場合にも，その力は操作シリンダで発生するので，てこの入力端に必要な力はきわめてわずかでよい。

電気式サーボ機構　電気式サーボ機構は，駆動部分に電動機を用いたもので，その電動機を制御するために，電力増幅器が用いられる。電気的信号は伝送に便利なため，目標値を与える場所と，サーボ機構の負荷のある場所とが離れている場合には有利である。一方，応答の速さの点については，油圧式のものに劣ると考えなければならないが，経済性，取り扱いやすさの点では，電気式の方が優れている。

図2は，電気式サーボ機構の例である。

図において，左端のハンドルの回転角 θ_r は目標値であり，右端の負荷の回転角 θ_c は制御量である。P_1, P_2 は同一特性をもつポテン

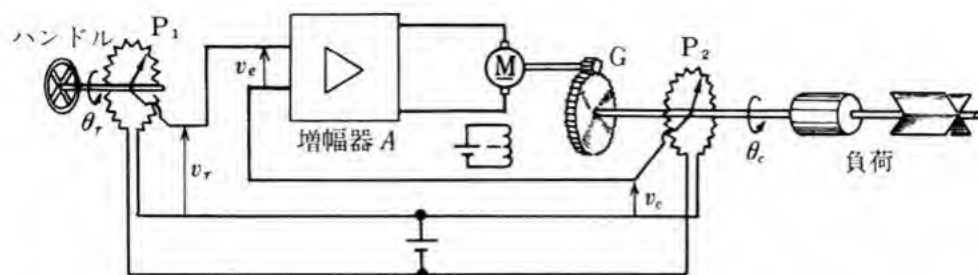



図 2　電気式サーボ機構の例

ショメータで，θ_r および θ_c をそれぞれの角度に比例した電圧 v_r, v_c に変換する，いわゆる角度-電圧変換器である。M は，駆動用の直流電動機で，ふつうサーボモータとよばれる。G は歯車機構である。一般に，サーボモータは，同一出力の場合，高速・低トルクの方が小形となるので，比較的高速のサーボモータを用いて，歯車で減速して，使用することが多い。

いま，左端のハンドルで目標値 θ_r を変えると，P_1 からの電圧 v_r と，P_2 からの電圧 v_c との間に差 v_e を生じ，これが増幅されて電動機に加えられ，負荷が駆動される。これは，v_e が 0 になるまで続き，v_r と v_c とが等しいところ，すなわち目標値 θ_r と制御量 θ_c とが等しいところで停止する。

このようにして，θ_c はつねに θ_r に追従する。

(3)　サーボ機構用機器

サーボ機構において，一般に使用されている機器について調べる。

検出器　サーボ機構において，制御量である位置または角度を検出するために使用される機器である。位置の検出用としては，ポテンショメータ（直線形）（図 3 (a)）・差動変圧器（図(b)）がよく用いられ，また角度の検出用としては，シンクロレゾルバおよびポテンショメータ（回転形）（図(a)）がよく用いられる。

操作用機器　制御対象を駆動する機器を操作用機器という。

（直線形）　（回転形）

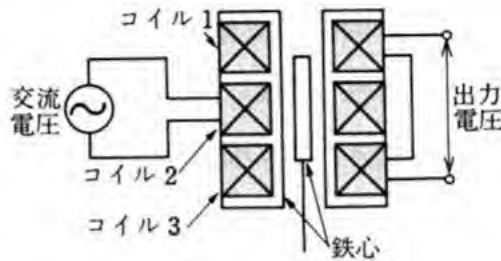
(a) ポテンショメータ

(b) 差動変圧器

図3 検　出　器

油圧式サーボ機構の場合は，操作用機器として操作シリンダが広く用いられる。操作シリンダは，図1に示したように，案内弁などと組み合わせて用いられる。

一方，電気サーボ機構においては，電動機が使用される。そのとき要求される特性としては，1) 低速時にも円滑な回転が得られること，2) 急加速・急減速が可能なこと，が挙げられる。このような要求を満たすように作られた電動機をとくに**サーボモータ**（servo motor）とよんでいる。サーボモータには，直流で駆動する**直流サーボモータ**と，交流で駆動する**交流サーボモータ**がある。

サーボ増幅器　電気式サーボ機構の場合，サーボモータを駆動するのに，増幅器が必要となる。この場合も，とくにサーボ機構に適する特性をもつように設計された増幅器をサーボ増幅器とよぶ。近年，とくに小容量のサーボ機構では，集積回路によるサーボ増幅

器がよく使用されている。

なお，油圧式サーボ機構の場合に，案内弁は増幅器と考えることができる。

（4） サーボ機構の使用例

ならい旋盤　旋盤やフライス盤において，削るべき形の型板を作っておき，材料をこれと相似形に自動的に削る方式を **ならい削り** といい，これにしばしばサーボ機構が用いられる。

図4は，油圧式サーボ機構を旋盤に応用した例である。

図において，主軸を回転しながら刃物台を左に動かすと，トレーサは型板に接しながら動き，その動きは案内弁のスプールの動きとなり，圧油が操作シリンダに流れて，バイトをスプールの動きと同じだけ動かすようになっている。

このような方式は，フライス盤にも適用でき，プロペラやタービンの羽根などのような複雑な形状のものを削るのに適している。

問 1. 図5の系は，サーボ機構として働くか。もし働かない場合，フィードバック用てこと，案内弁・操作シリンダとの結合はそのままとし，どのように改良すればよいか。

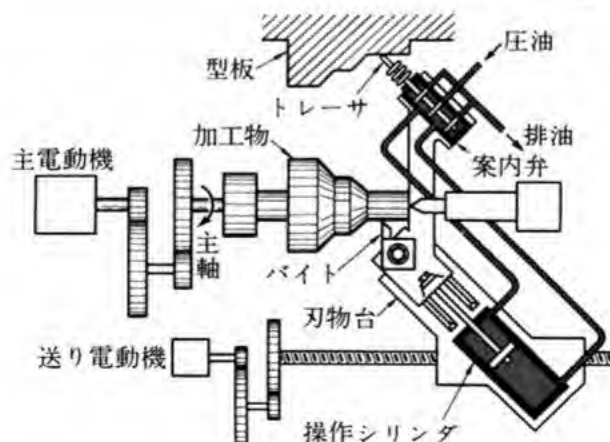



図4 ならい旋盤

問 2. 図1と図2のサーボ機構を比較して，大きく異なる点を三つ挙げよ。

図 5

2. 数値制御

(1) 数値制御

先に学んだならい旋盤を用いて加工する場合，一回の切削で製品が得られるわけではなく，粗・中・仕上げの各加工段階を経なければならない。したがって，それぞれに対応する型を用意しておく必要があり，さらに各加工段階の間で，型の取り換えを行う必要がある。型の作成は，とくに仕上げ加工用のものは，高精度でなければならないので容易でなく，また型の取り換えの時間も必要である。

これに対して，**数値制御**（numerical control: **NC**）は，粗・中・仕上げなどの各加工段階での加工順序・加工寸法・加工速度など，加工に必要な情報を記憶させた指令テープを作成しておき，加工を行う場合には，その内容を**テープ読取り機**（tape reader）で読み取り，これを解読しながら，定められた順序に，工具の位置や動く経路や速度を制御して，自動的に加工を進める方式であり，シーケンス制御とフィードバック制御との両方が含まれている。数値制御の利点の主なものを挙げると，次のようになる。

1)　加工精度を上げることができる。

2)　1本の指令テープに，加工に必要なすべての情報を記憶させることができるので，全加工終了まで人手を省くことができ，かつ一様な品質の製品を生産することができる。

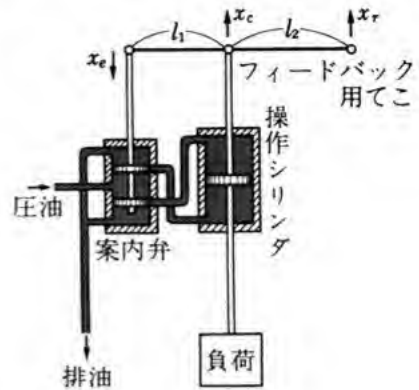
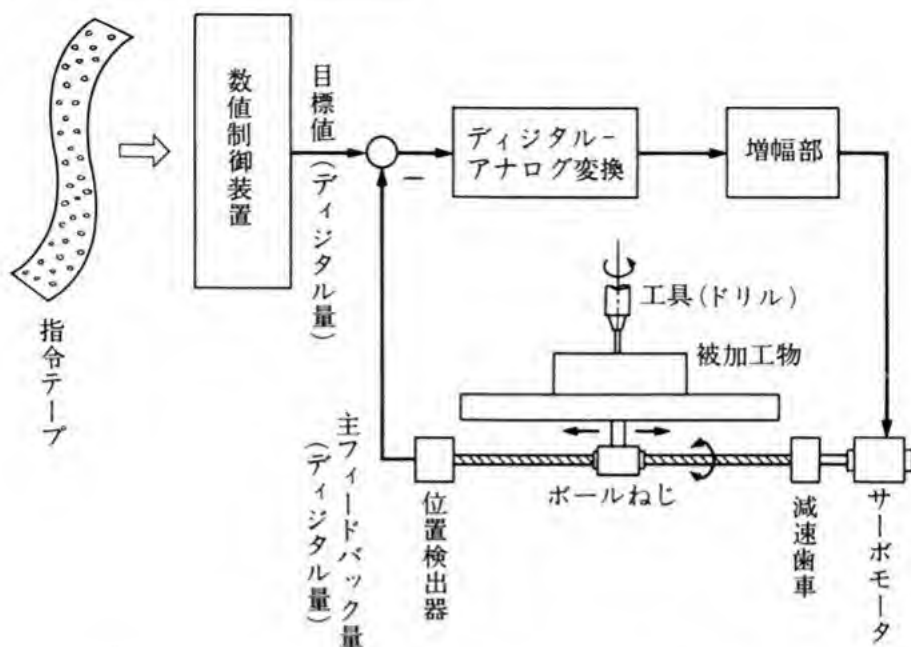



図 6　数値制御の構成

3)　指令テープの作成は，型の作成に比べて一般には容易である。

最近では，数値制御は，工作機械だけでなく，産業用ロボット・搬送装置など，工場での加工・組立て・搬送全般にわたって，広く利用されている。

（2）　数値制御の基本的構成

図6は，数値制御の代表的な構成を例によって示したものである。

この例で，数値制御装置から送られる目標値は，ディジタル量であり，これによってサーボ機構が動く。そのさい，目標値がテーブルの指定位置そのもので，したがって，主フィードバック量も実際のテーブルの位置をディジタル量で検出したものである場合，この方式を **絶対座標方式** とよんでいる。これに対して，与える目標値がテーブル位置の変更量として指定したものであり，したがって，主フィードバック量は実際のテーブルの移動量をディジタル量として検出したものである場合，この方式を**増分**（ぞうぶん）**方式（インクレメンタル方式）**

とよんでいる。

機械加工には，穴あけのためのドリルや被加工物などの**位置決め制御**と，形を削り出すための**輪郭制御**とがある。ここでは，以下，位置決め制御について説明する。

図6で示すように，被加工物を決められた位置に動かし，穴をあける場合，移動速度や移動の経路（二次元の位置決めの場合）はあまり問題ではなく，ドリルの先端位置にあけようとする穴の位置を正確に合わせることが目的となる。

増分方式の場合，まずテーブルを現在位置から移動させる距離を指令ディジタル量として与え，これをレジスタに記憶させておく。次に，テーブルが単位距離だけ移動するごとに，変位検出器より一つのパルスを発生させてフィードバックし，カウンタで計数し，これが指令ディジタル量と一致したとき，テーブルを停止させる。ここで用いられる検出器としては，正方向・負方向にそれぞれ対応したパルスを発生させるパルス発生器が用いられる。この方式は，装置が簡単であるが，計数の誤り（ミスカウント）がそのまま誤差として残ることに注意する必要がある。

これに対して，絶対座標方式では，テーブルの固有の原点を決めておき，これから見た目標位置を指令ディジタル量として与え，これと，検出されたテーブルの現在位置との差を減算器で求め，これが0になったとき，停止させる。この場合の検出器は，パルス発生器に比べると複雑で高価であるが，ミスカウントの問題がない。

問 3. 数値制御のもつ利点について述べよ。

問 4. 増分方式と絶対座標方式との違いを説明せよ。

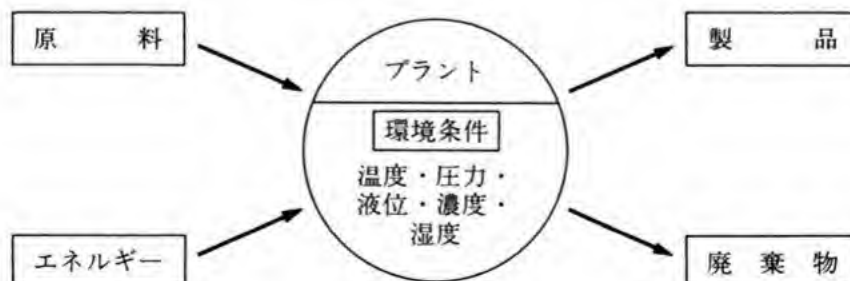
3. プロセス制御

(1) プロセス制御

化学工業・石油工業・製紙工業または火力発電などのようないわゆるプロセス工業では，図7のように，所定の原料とエネルギーとを製造装置，すなわち，プラントに供給して，それらに関する温度・圧力・液位・濃度などの環境条件を，定められた値に保つことによって，目的の製品が生産される。したがって，均一な製品を作るためには，これらの環境条件および原料・製品・廃棄物の流量や，エネルギーの量などについて制御を行わなければならない。

例えば，火力発電におけるボイラの制御では，原料としては水，エネルギーとしては重油・空気，製品としては蒸気，廃棄物としては煙突からの廃棄ガスがあり，制御しなければならない物理量としては，蒸気圧力および温度，ドラムの水位，燃焼室内の圧力および原料・エネルギーの流量などがある。

このように，一般に **プロセス工業** では，多くの制御量が含まれ，これらを同時に制御しなければならないので，プロセス制御系の構成は，サーボ機構に比較して大規模で複雑である。

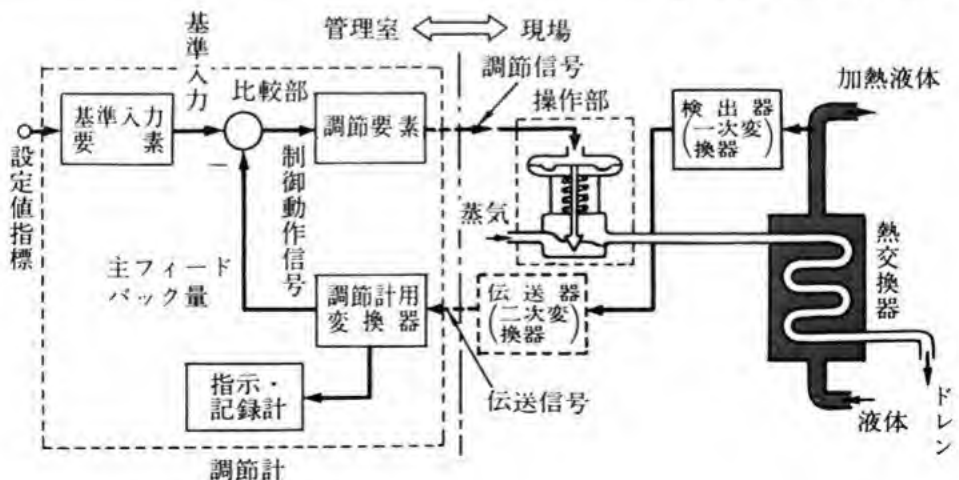
図7 プロセス工業

(2) プロセス制御系の構成

ここでは，比較的基本的なプロセス制御系の構成について学ぶ。図8は，その一例で，熱交換器から出る液体の温度を，熱交換器に送る蒸気量によって制御しているプロセス制御系である。

図からわかるように，プロセス制御系は，プラントのある現場と，管理室とにまたがって構成されており，管理室には，主に制御を行うための頭脳に相当する部分が収められている。

まず，制御対象より制御量（温度）を**検出器**で検出する。この信号は，比較的離れたところにある管理室に伝えなければならないので，**伝送器**（transmitter）により増幅され，伝送に都合の良い量に変換される。管理室に伝送された信号は，**調節計**（recording controller）に加えられる。調節計の中では，次のように信号が伝わる。

まず，伝送されてきた信号は，目標値と比較するのに都合の良い量に変換し，主フィードバック量として**比較部**に伝送される。比較の結果，得られた制御動作信号は，調節要素に送られる。
p. 170で特性改善について学んだように，調節要素では，系全体が良好な結果が得られるような調節信号を発生し，この信号は，再び

図 8 プロセス制御系の構成

現場にある**操作部**に伝えられる。

なお，調節計に加えられた信号は，必要に応じて，**記録計・指示計**によって，記録あるいは指示される。

(3) プロセス制御用機器

検出器　プロセス制御における制御量としては，圧力・流量・温度などが考えられる。これらの制御量を検出する方法について考える。

圧力検出用に用いられる圧力計は，多くの場合，圧力を変位に変換する要素からなっている。図9は，**弾性圧力計**の原理と，測定可能な範囲を示したものである。

流量検出用に用いられる流量計には，絞り流量計と面積式流量計とがある。**絞り流量計**は，図10のような構成をしており，絞りの上流と下流との間に生じる圧力差が流量の2乗に比例することを利用している。これによって，流量が圧力差に変換できるので，この圧力差を検出することによって，流量を検出することができる。圧力差の検出の一例としては，U字管を用いた図10のような方式がある。

そのほか，温度検出には，電気抵抗の温度変化を利用した抵抗温度計，異種金属の接合点における起電力の温度変化を利用した熱電

図9 弾性圧力計

(a) ベロー (0〜3[kgf/cm²])　(b) ダイアフラム (0〜300[mmH₂O])　(c) ブルドン管 (0〜1000[kgf/cm²])

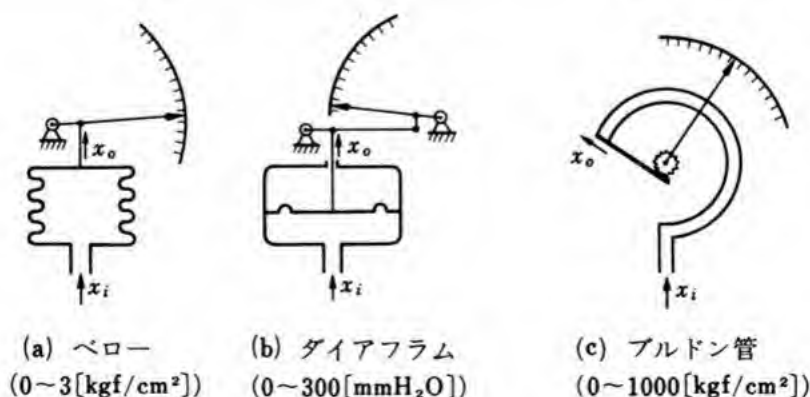
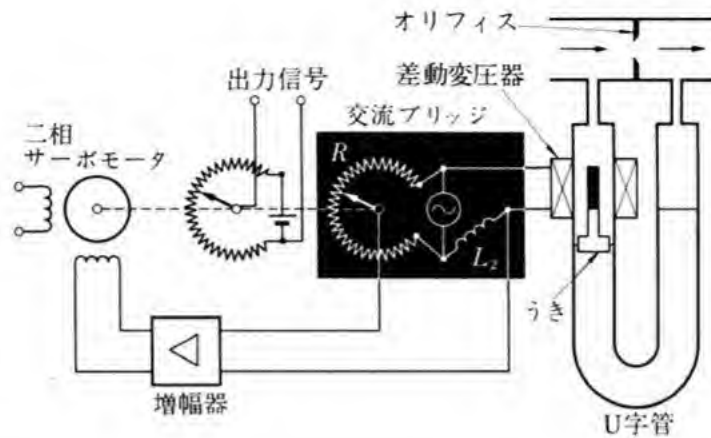



図 10　絞り流量計

温度計などがある。

伝送器　検出器からの信号が微弱な場合，または伝送に適さない場合，これを増幅したり，伝送に便利な信号に変換しなければならない。とくに，大規模なプラントにおけるプロセス制御系の場合，現場から離れて集中管理室を設け，そこに調節計をすべて集中して設置することが多い。このような場合に，信号の伝送の役目を果たすものが伝送器である。伝送器には，信号を空気圧の形で伝送する空気式伝送器と，信号を電気量に変換して電気信号として伝送する電気式伝送器とがある。図 11 は，**空気式伝送器**の一例である。

調節計　調節計は，図 8 に示したように，基準入力要素，調節計用変換器，比較部および指示・記録計からなりたっている。

このうち，主要な部分である調節要素の役割は，比較部で求められた制御動作信号から調節信号を作って，操作部に送ることである。このさい，比例要素・微分要素・積分要素を用いて，単に制御動作信号を増幅するだけでなく，信号の微分値，積分値を用いて，制御系の制御特性を良くする役割をもっている。図 12 は，**電気式調節計**の外観を示したものである。

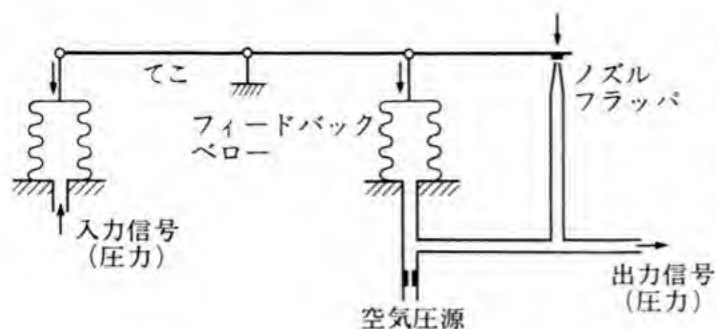
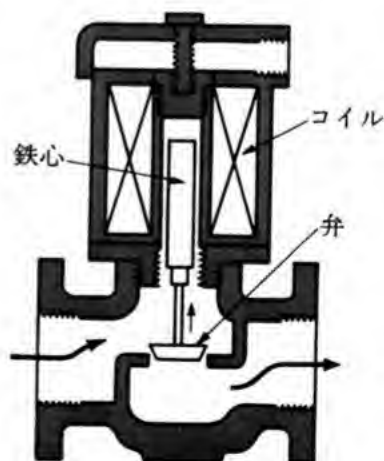
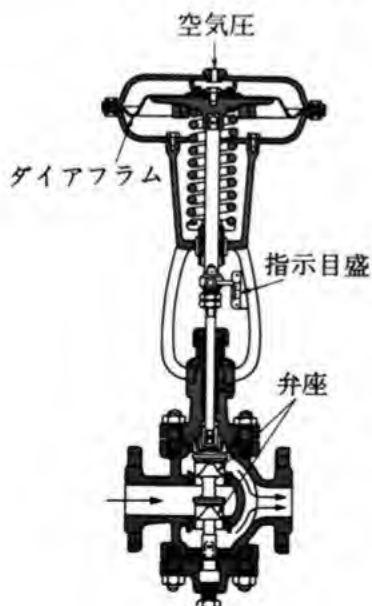



図 11 空気式伝送器 図 12 電気式調節計

操作部 操作部は，調節要素から調節信号を受け，制御対象を直接制御する部分である。プロセス制御では，ふつう，弁の開閉によって，調節信号に比例したエネルギー・原料などの流量を変化させる。

弁には，空気式・電圧式・油圧式などがある。図13は，空気圧を用いた**ダイアフラム弁**であり，図14は，**電磁弁**の構造を示したものである。

図 13 ダイアフラム弁 図 14 電 磁 弁

問 5. プロセス制御計の基本的な構成について説明せよ。

問 6. 調節計の役割について説明せよ。

問　　題

1. 図15において，R_1, R_2 は既知の抵抗で互いに等しい値をもち，また，R_4 は，可変抵抗器でその値がダイヤルで読み取れるようになっているとする。いま，抵抗 R_3 が未知の値であるとして，サーボ機構を利用してその値を自動的に求める方法を考案せよ。

2. サーボ機構が工作機械類において，どのような形で利用されているか調べよ。

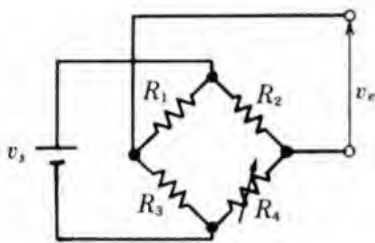
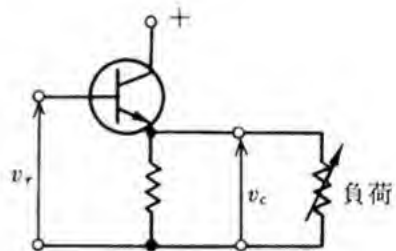



図 15　　図 16

3. 図16はエミッタホロアである。図で電圧 v_r を一定とすると，負荷抵抗の変化に対して v_c はあまり変動しないことをフィードバック制御と同じ考え方にたって説明せよ。

4. プロセス制御系における伝送器の役割について述べよ。

5. 図17のように，台の上に容器を置くと，水が定量だけ自動的に注がれるようなシーケンス制御を考案せよ。ただし，入力端子に電流を流すとバルブが開くような電磁弁が利用できるものとする。

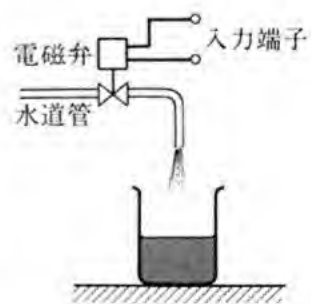



図 17

問　題　解　答

第10章　電子計算機

2　節

問 1.　11, 50　**問 2.**　$(6.375)_{10}$　**問 3.**　$(1110111.011)_2$

問 4. (1)　27　(2)　18　(3)　13.5　(4)　10.5　(5)　14.5

問 5. (1)　28　(2)　70　(3)　21

問 6.　$(181)_{16}$　**問 7.**　(1)　$(110000011)_2$, $(183)_{16}$　(2)　$(1100001101)_2$, $(30D)_{16}$　(3)　$(1101111.1)_2$, $(6F.8)_{16}$　(4)　$(1011101.11)_2$, $(5D.C)_{16}$　(5)　$(0.101)_2$, $(0.A)_{16}$

問 9.　6ビット

問題 1.　(1)　1100　(2)　0010　(3)　1101110010　(4)　1011

2.　(1)　54　(2)　40.75　(3)　11.25　(4)　15.625

3　節

問 1.

A	B	$A \cdot B$	C
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	1

A	B	$A+B$	C
0	0	0	0
0	1	1	0
1	0	1	1
1	1	1	1

問 6.　2^2, 2^1, 2^0 の各端子

4　節

問 3.　630バイト

5　節

問 7.　(1), (4)　実数型　(2), (3)　整数型

問 8.　(1)　A＊X＊＊2＋B＊X＋C　(2)　(B＋C)/(2＊A＊＊2)
(3)　A/((B＋C)＊＊2)　(4)　SQRT(A＊＊2−R＊＊2)

第11章　自動制御

2　節

問 8.　(a)　$(B+C)\cdot A=X$　(b)　$A\cdot C+B\cdot\bar{D}=X$

問題 2.　$(A\cdot B+C)\cdot D=E$

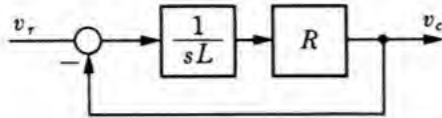
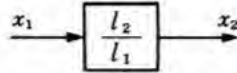
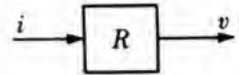
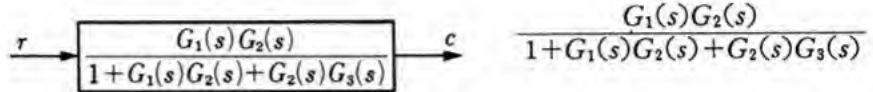
3　節

問 3.

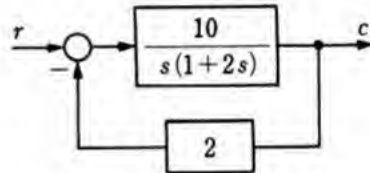
問 4.

問 5.　$G=R$

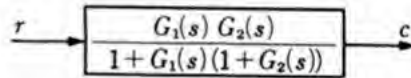
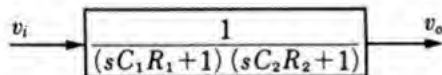
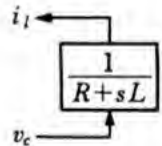
問 6.

$$\frac{G_1(s)G_2(s)}{1+G_1(s)G_2(s)+G_2(s)G_3(s)}$$

問 7.　$\dfrac{1}{1+s\dfrac{L}{R}}$　　問 8.　$T=0.5$〔s〕

問 9.　$\dfrac{\dfrac{R_eR_l}{R_f}}{R_l+R_i}\cdot\dfrac{1}{1+s\dfrac{L_f}{R_f}}$

問 10.

問 11.　$M(s)=\dfrac{5}{s^2+0.5s+10}$，$\omega_n=3.16$，$\zeta=0.079$

問題 1.　$\dfrac{1}{R}+sC$　　2.

3.

4.　$C=2$〔μF〕

5.　p. 154 の図 18 (b) に右のブロックを付け加えればよい。

6.　$M(s)=1+sRC$

4 節

問題 1.　v_e を制御動作信号，R_4 のしゅう動子位置を制御量とするサーボ機構

索　　引

け

こ

さ

し

■別記著作者

長谷川健介

矢田　光治

田中　　隆

石井　孝司

●表紙・扉AD
多川精一　＋　中野達彦
（東京エディトリアルセンター）

●表紙
飯田　敏行

電気技術II C

7	実教	工業　067

昭和57年３月31日　文部省検定済
昭和58年２月25日　初版発行
平成２年１月20日　印　　刷
平成２年１月25日　発　　行

©著作者——●
大河内正陽　ほか４名（別記）

発行者——●
実教出版株式会社
代表者　奥脇　誠治
東京都千代田区五番町５

印刷者——●
中央印刷株式会社
代表者　日岐　弘登
東京都新宿区新小川町４－24

発行所——●
実教出版株式会社
〒102　東京都千代田区五番町５
電話　03－238－7700（代表）
振替　東京　4－183260

定　価——●
文部大臣が認可し官報で告示した定価〔消費税に相当する金額を含む〕（上記の定価は，各教科書取次供給所に表示します）